AMERICAN UNIVERSITY OF BEIRUT

# Optimal Path Planning for a Class of Nonholonomic Systems with Drift

by

Bilal Abdel Nasser Hammoud

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Mechanical Engineering
of the Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
November 2016

AMERICAN UNIVERSITY OF BEIRUT

# Optimal Path Planning for a Class of Nonholonomic Systems with Drift

by

Bilal Abdel Nasser Hammoud

Approved by:

Prof. Elie Shammas, Assistant Professor — Advisor
Mechanical Engineering

Prof. Daniel Asmar, Associate Professor — Member of Committee
Mechanical Engineering

Prof. Imad Elhajj, Associate Professor — Member of Committee
Electrical and Computer Engineering

Date of thesis defense: November 23 , 2016

**AMERICAN UNIVERSITY OF BEIRUT**

**THESIS, DISSERTATION, PROJECT RELEASE FORM**

Student Name: Hammoud Bilal Abdel Nasser
Last First Middle

☑ Master's Thesis ◯ Master's Project ◯ Doctoral Dissertation

☑ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after : **One ---- year from the date of submission of my thesis, dissertation, or project.**
**Two ---- years from the date of submission of my thesis, dissertation, or project.**
**Three ---- years from the date of submission of my thesis, dissertation, or project.**

Bilal Hammoud Nov, 30, 2016

Signature Date

This form is signed when submitting the thesis, dissertation, or project to the University Libraries

# Acknowledgements

The most difficult part of writing this thesis is, with no doubt, this part. I would like to express my gratitude and thankfulness for everyone who helped make writing this thesis possible.

First and foremost, I would like to express my sincere gratitude and appreciation to my thesis advisor *Elie Shammas* for his continuous support, great advice when most needed, and for being the person who most influenced my experience here at AUB. I will always owe him for what I achieved so far and will achieve in the future as I could have not asked for a brighter and more humble mentor.

None of this was possible without the endless support of my family. For this I will never forget the encouragement and tolerance of my parents *Abdel Nasser* and *Fatima*, my brother *Hamza* and my sister *Somaya*. To my uncle *Hisham*, thank you for the all the help which, without, I would have never got here.

I would like to thank *Dr. Daniel Asmar* for introducing me to the Vision and Robotics lab and for reminding me that I can always do better. I owe *Dr. Imad Elhajj* for his insightful comments and questions the helped shape this thesis throughout the past two years. I would also like to mention what a great teacher *Dr. Naseem Daher* was, and how he is the most committed and dedicated person I ever met.

I can't forget the members of the Vision and Robotics Lab for putting up with all my distractions especially *Salah Bazzi*, *Mohammad Al Salman*, *Rahaf Rahal* and *Mahmoud Hamandi*. I would like to thank *Ali Harakeh* for showing me what hard work is really like. I am also thankful to *Abdel Rahman Al Makdah* for constantly reminding me that things could always be much worse. One last person I would like to thank is *Sevag Babikian* for being a great friend and a good teacher.

# An Abstract of the Thesis of

Bilal Abdel Nasser Hammoud for Master of Engineering
Major: Mechanical Engineering

Title: Optimal Path Planning for a Class of Nonholonomic Systems with Drift

For a system to be completely autonomous, many subsystems need to interact and ensure that the system executes the required tasks without any human intervention or guidance. One subsystem that computes the trajectory which an autonomous agent should traverse while going from an initial configuration to a predetermined final configuration is an essential part to ensure complete autonomy. Solving the path planning problem is a major research area in the field of autonomous robotics. Researchers tend to use simplified kinematic models to obtain solutions, frequently analytic, for the planned trajectories of autonomous agents.

This thesis introduces a new kinematic model to describe the planar motion of an Autonomous Underwater Vehicle (AUV) moving in constant current flows. The AUV is modeled as a rigid body moving at maximum attainable forward velocity with symmetric bounds on the control input for the turning rate. The model incorporates the effect a flow will induce on the turning rate of the AUV due to the non-symmetric geometry of the vehicle. The model is then used to characterize and construct the minimum time paths that take the AUV from a given initial configuration to a final configuration in the plane. Two algorithms for the time-optimal path synthesis problem are also introduced along with several simulations to validate the proposed method.

After that the assumption of maximum forward velocity is relaxed. Then a new type of paths is investigated. The trajectories computed allow the system to travel between two predetermined configurations while minimizing the total power consumption. Finally, the results are obtained numerically and geometric interpretations are presented.

# Contents

# List of Figures

*You will come to know that what appears today to be*
*a sacrifice will prove instead to be the greatest*
*investment that you will ever make.*
*Unknown*

# Chapter 1

# Introduction

The reliance on machines that operate on their own without any human interaction is increasing. Examples of such robots include, but not limited to, the Roomba Vacuum Cleaning Robot, the Google Self Driving Car and surveillance drones. The development of such systems is still in its early stages and it is a long way before completely autonomous robots will be able to take a part in our daily life and operate safely without the use of any human guidance or control.

A completely autonomous robot will comprise of many subsystems working together. A high level mission planning system is required to plan the tasks a robot is supposed to execute. A path planning algorithm then generates the states that the robot must track during a certain time interval based on its current state in order to execute the desired mission or task. A low level controller then controls the actuators to ensure that the robot tracks the desired trajectories generated by the path planning algorithm. In order to close the feedback loop, perception is achieved through a collection of sensors that measure certain outputs of the robot, such as encoders for angular or rotational measurements, inertial measurement units for acceleration readings, GPS for coordinate measurements and many other sensors. However, it is not possible to install sensors to read all the states needed, so an observer or a state-estimator is needed to estimate the remaining states based on the readings of the sensors. A simple block diagram presenting the basic building blocks of an autonomous robot is shown in Fig.1.1. The robot's estimate of its current state could be used as the starting or initial configuration for the planning algorithm in order to generate the trajectory connecting it to the final configuration desired.

## 1.1 Literature Review

This work will focus mainly on path planning for autonomous robots. Planning algorithms such as the $A^*$ algorithm [1], Rapidly-Exploring Random Trees

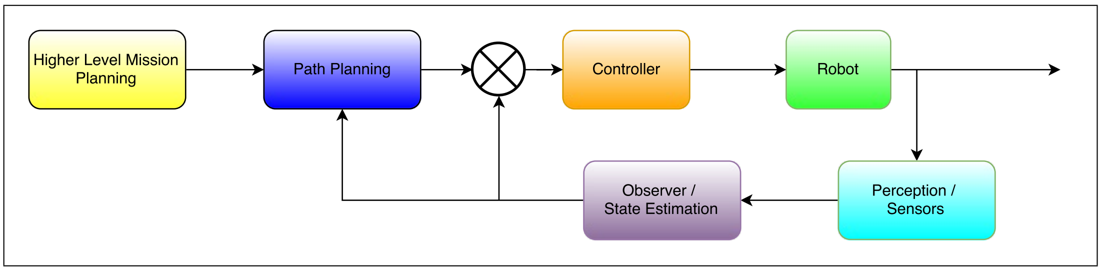



Figure 1.1: The basic building blocks of an autonomous robot

(RRT) [2], and many others presented in [3] use waypoint planning approaches. Such approaches perform well for robots that are not constrained in motion, also known as holonomic robots. However, for robots with nonholonomic constraints the approaches presented earlier are not sufficient and would normally produce trajectories that are not feasible or cannot be traversed by nonholonomic robots. For a path planning algorithm to generate feasible trajectories for nonholonomic robots, a thorough study of the constraints imposed on the system is required, only then a planning algorithm can be devised to take into consideration the specific nonholonomic constraints defining the type of movements that a certain robot can execute. Taking the nonholonomic constraints into consideration further complicates the path planning problem. Holonomic and nonholonomic systems and robots are described in more details later in chapter 2.

The class of nonholonomic systems of interest in this work includes systems possessing similar constraints to that of a regular car but traveling in a medium with a constant drifting field. This work is motivated by the Hybrid Autonomous Underwater Vehicle $(AUV)$ shown in Fig.1.2. This AUV doesn't have the capability of rotating in its position or moving in a lateral or a sideways direction, very similar constraints to that of a regular car. This work can also be applied to Unmanned Aerial Vehicles flying in wind fields or for ships and surface vehicles traveling in the presence of currents.

The behavior of such systems was modeled on the kinematics level in the late 1880's where Markov introduced the following minimum time problem in [4]: given any initial and final positions and orientations in a plane, what is the minimum length trajectory given that there exists an upper and lower bound on the curvature of the curve? In 1957, L. E. Dubins [5] presented his solution to the Markov problem showing that a solution exists and comprises of two maximum turning rate segments joined by a straight line, or three maximum turning rate segments if the Euclidean distance between the initial and final positions is less than four minimum turning radii. While Dubins solved the problem using geo-

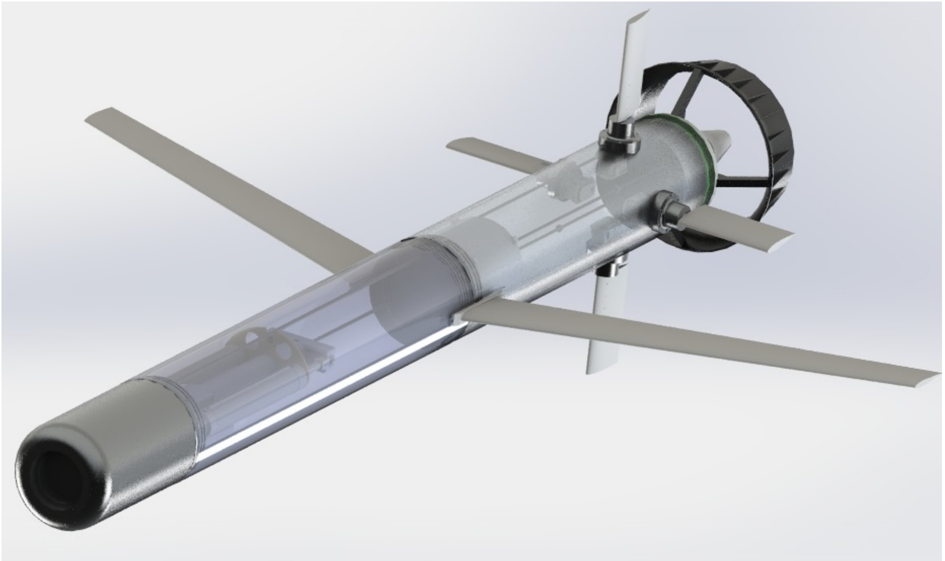
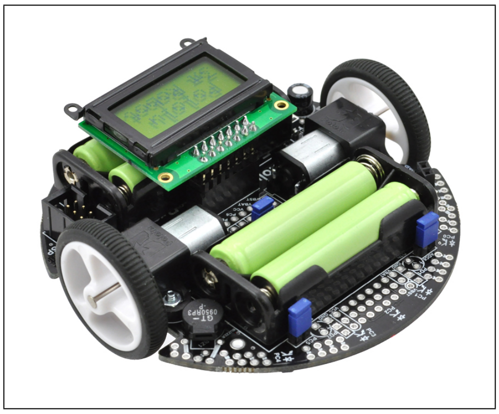


Figure 1.2: Hybrid Autonomous Underwater Vehicle

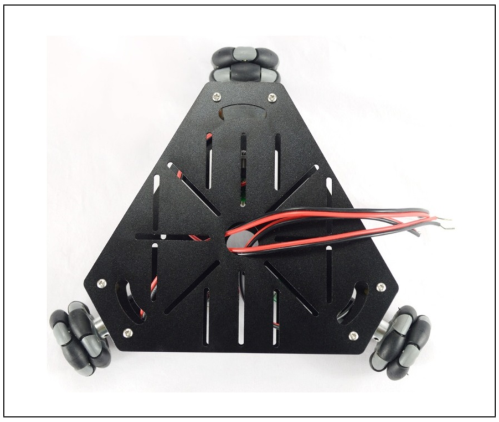
metric arguments only, Boissonnat et al [6] solved the minimum time problem by using an optimal control approach. The complete path synthesis problem for the Dubins car model was solved by Bui et al [7].

Many variations of the Dubins-Markov problem were presented later on. Reeds and Shepp [8] solved the minimum time problem for a Dubins car that goes both forwards and backwards, thereby introducing cusps to the solution presented by Dubins. Soueres and Laumond [9] later developed a complete study to provide a method of selecting the proper path connecting any two configurations.

With most of the minimum time path planning work being done on car like robots, Balkcom [10] presented the minimum time paths for a differential drive robot similar to the robot shown in figure 1.3a in an obstacle free plane. Balkcom [10] also proved the existance of optimal controls. After 40 different classes of candidate time optimal trajectories were identified. Finally and algorithm to generate time optimal trajectories between any two configurations is provided. In [11] Balkcom also investigated the minimum time problem but for an omni-wheeled robot, shown in figure 1.3b. For a regular omni-wheeled robot it is clear that the shortest path is a straight line, Balkcom studied omni-wheeled robots that can move more quickly in some directions than others. The work presented in [11] provided analytical solutions for the fastest trajectories between any two configurations. The time optimal trajectories for the omni-wheeled robot comprise of spins in place, circular arcs and straight lines parallel to the wheel axles. Balkcom then identified four classes of candidate concatenations of time-optimal segments.



(a) Pololu 3PI Differential Drive Robotic Platform



(b) Nexus Robots Omni-wheeled Robotic Platform

Figure 1.3: Examples of Available Wheeled Mobile Robotic Platforms

A generalization of Dubins and Reeds and Shepp trajectories was presented by Furtuna and Balkcom in [12]. Furtuna described the sequence of rotations and translations required to generate a minimum time trajectory connecting two configurations of a rigid body belonging to *SE(2)*. Furtuna then developed an algorithm that generates a finite set of structures of optimal paths given a set of optimal segments or controls.

In [13] Chitsaz and LaValle introduced the Dubins airplane model extending the Dubins car model into *3D* space by assuming an independent control over the altituted. Then with the help of Pontryagin's Maximum Principle the optimal segments building up the optimal trajectories came out to be similar to that of Dubins car along with pieces of planar elastica added to them. McGee and Hedrick [14] introduced a kinematic model for an airplane moving in constant wind currents in *SE(2)*. The model introduced is similar to a Dubins car model with constant drift terms added to the translational components of the model. McGee and Hedrick then proved that the set of Dubins trajectories is incomplete in the presence of wind. To get the complete solution McGee and Hedrick formulated the problem as a virtual moving target problem then presented an iterative method to get the complete solution. Later, Bakolas and Tsiotras [15] considered a combination of the Markov-Dubins problem and the Zermelo problem to study the motion of an airplane in constant wind fields. They exploited the optimal control structure to show that even though the control structure is similar to that of a Dubins problem, the synthesis problem turns out to be significantly different. Considering time varying wind currents or vector fields, McNeely [16] proved the existence and uniqueness of minimum time trajectories and then introduced an algorithm for obtaining the minimum time paths for such vehicles.

Rysdyk [17] explained that an airplane moving in a constant flow field and executing maximum turning rate curves would traverse a trochoidal path. Using the results obtained by Rysdyk, Techy and Woolsey [18] derived analytical solutions for the minimum time path planning problem of a planar airplane in a constant flow field. Many variations of the problem described earlier were extensively studied later on. For example, Bakolas and Tsiotras [19] considered a Dubins car model having non symmetric bounds on the steering control input. The structure of the optimal control is the same as that of the Dubins model but the different turning radii to each side affect the synthesis problem and furthur complicate it. Choi [20] also studied Dubins car and airplane models belonging to *SE(2)* while experiencing severe damages or control failures. In the presented cases a unidirectional constraint is imposed on the controls and the lower and upper bounds on the turning rate take the same sign, meaning that the vehicle is always rotating to one side at a turning rate $\in [a, b]$ where $a$ and $b$ are of the same sign. Choi then solved the minimum time path planning problem and extended the planar results to the planning of aircraft emergency landing.

On the other hand, and inspired by sea vessels traveling in ocean currents, Dolinskaya [21] presented a model where the direction dependence of the minimum turning radius and the maximum forward velocity of a ship traveling in ocean currents is considered. Later, Dolinskaya and Maggiar [22] characterized the time optimal trajectories of a vessel traveling in anisotropic *(direction-dependent)* environments where a more generalized model of Dubins car is used to show that there exists an optimal path that is a subset of a path of the type *CSCSC* where $C$ denotes a maximum turning rate curve to the left or right and $S$ represents a straight line. At the same time, Chang [23] developed similar work, where the minimum turning radius of robots working in mines depends on the slope and nature of the ground making it direction-dependent. In both cases, [22] and [23], the minimum time path planning problem was addressed, and a general characterization of the optimal path structures for general direction-dependent models was discussed.

The optimization criterion considered in all the work presented previously was the total time of travel between two configurations. Another optimization criterion considered by researchers earlier is the total energy expenditure or the total control effort required to travel between two predefined configurations. In [24], Walsh et al. studied a particular problem known as the *Landing Tower Problem*. The authors of [24] formulated the problem as an optimal control problem on *SE(3)*, then the solution is presented using Pontryagin's maximum principle and numerical optimization methods. The system is presented as a left invariant control system on *SE(3)* and the cost minimized is defined as the total control expenditure along the path traversed.

Mukherjee and his colleagues considered the Dubins car model in [25] with the aim of generating minimum control trajectories. First using Pontryagin's maximum principle the problem is converted into a two-point boundary value problem and solved using numerical relaxation methods available for such problems. The authors then point out similarities between the energy-optimal motion of the two-wheeled mobile robot and the motion of a pendulum in a gravitational field. An analytical solution for the minimum energy paths is obtained in terms of *Jacobi Elliptic Functions*. Later, Duleba and Sasiadek [26] presented a modification of the well know *Newton Algorithm* for the aim of generating minimum energy trajectories for a general class of *driftless* nonholonomic systems.

Mei et al. presented in [27] a new approach for finding energy efficient trajectories for omni-wheeled robots, where the motors speed is related to their power consumption through a polynomial map. The velocity of the robot is then derived as a result of a linear transformation of the wheels' velocities. After that the energy consumption for different planning methods was studied as a function of the area being scanned by the mobile robot. Finally the results obtained were validated through experimentation.

Few years later Moll and Kavraki used similar approaches to solve path planning problems for flexible wires used in surgeries. In [28] the authors parametrized low energy configurations corresponding to the flexible wire. Then the curves satisfying specified end constraints for the fixed length wires were presented. Finally the authors developed three different methods for optimizing the energy within the wire. Moll and Kavraki presented atfer that a new approach in [29] for planning trajectories that guarantee stable configurations for the flexible wires. These stable configurations correspond to minimum energy curves that are parametrized by the authors. The number of parameters in the work presented depends on the complexity of the curves. The planning algorithm presented automatically varies the number of parameters while computing paths from one minimum energy curve to another minimum energy curve.

The energy optimal paths for a *differential driven* wheeled mobile robot were studied by Kim in [30]. The author built his formulation based on Dubins car model but including independent controls on both forward velocity and steering. First he derived a closed form solution for the minimum energy control input along a straight line trajectory. Next the author investigated the turning trajectories corresponding to minimum energy maneuvers. Finally an iterative search method was used to find a trajectory that comprises of the derived segments such that it minimizes the total cost of the robot's battery energy consumption. Spangelo and her co-authors [31] used numerical techniques to plan energy optimal paths for a solar powered aircraft in a loitering mission. Periodic splines were first defined then the other parameters related to the aircraft were optimized.

Tokekar, Karnad and Lsler aimed in [32] to minimize the battery energy consumption for car like robots when traveling along a given path. To achieve this goal the authors modeled the energy consumption of the robot's DC motors. Then closed form solutions were presented for the case when the robot's maximum velocity is unbounded and the case where the velocity is bounded. Finally a calibration method was presented to estimate the model parameters and experiments were conducted on a costume built robot to validate the results obtained. Later, Tokekar and his colleagues studied in [33] the problem of finding energy optimal paths and velocity profiles to minimize the battery energy consumption during a mission for a car like mobile robot. First the problem of finding an energy optimal velocity profile for a given path was studied. Closed form solutions for the energy optimal velocity profiles were derived for a desired path. After that, the analytic solutions of the velocity profiles were used as subroutines in a more general algorithm to find the minimum energy trajectories. Finally, the results of experiments conducted for following optimal velocity profiles were presented.

In [34], Maclean and Biggs approached the problem of finding energy optimal trajectories for a simple wheeled robot modeled as a Dubins car using a completely different method. The authors first formulated the problem as a coordinate free optimal control problem on *SE(2)*. Then, an *invariant function* also known as a *Casimir function* was derived. The surfaces formed by the *control hamiltonian* and the *Casimir function* come out as a quadratic surfaces. The authors pointed out that the intersection of such surfaces forms an elliptic curve. Maclean and Biggs parametrized the control inputs in terms of *Jacobi elliptic functions* then a convex optimization method was used to solve for the parameters defining the control inputs given certain boundary conditions (initial and final configurations of the robot). Recently, Kularatne [35] developed a method for both time and energy optimal path planning for autonomous marine robots. The kinematic model used was similar to that used by Techy [18] and others. However Kularatne presented a graph search based method to solve the path planning problem for general flows.

## 1.2 Problem Statement

Most of the work done previously considered the agent as a particle moving either in a *2D* plane or in a *3D* space with a heading angle defined. Considering the system as a particle ignores any interaction between the system and the surrounding environment other than translational effects due to surrounding winds or current flows. Given a start and an end configuration in *SE(2)* find a time optimal trajectory as well as an energy optimal trajectory that connect two provided configurations such that there exists a constant flow affecting both the position and orientation of the robot and provided that the robot has bounded control inputs, namely its velocity and steering rate.

## 1.3 Thesis Outline

This thesis builds upon prior work and is closely related to the work done by Boissonnat [6], Techy [18], Dolinskaya [22] and many others. In Chapter 2 necessary background material that this work heavily relies on is introduced. A brief discussion of holonomic and nonholonomic constraints is carried out. Dubins car model and its later modification to incorporate constant currents are then introduced. After that, basic results in optimal control theory are presented including *Pontryagin's maximum principle*. Finally, numerical methods used to solve boundary value problems that typically emerge from the use of *Pontryagin's maximum principle* are discussed.

In Chapter 3 the motivation for developing a new model is discussed. Then the intuition behind the new model is introduced. After that the model is derived. Finally simulations are carried out to present the importance of the new model and compare its results with previously developed models.

Chapter 4 presents an optimal control approach to formulate the minimum time path planning problem. Applying *Pontryagin's Maximum Principle* to the problem defined makes it possible to derive discrete optimal control input values. Next, and with the help of geometric and analytical arguments, a proof relating the discrete control values to fixed geometric shapes is presented. After that three different families of possible concatenations are defined based on the work done by Dolinskaya [22]. A condition is then imposed on the speed of the flow field to limit the possible concatenations into two families only. Then two algorithms to construct all possible trajectories belonging to the two families of trajectories are presented. Finally simulations are carried out to show the resulting trajectories in comparison with previous models used for such problems.

Finally chapter 5 reformulates the optimal control problem into a minimum control effort or minimum energy expenditure problem. First, various numerical approaches to solve the resulting two point boundary value problem are presented. After that bounds on the costates of the system are derived and geometric interpretations are provided. Finally future research directions are presented based on the results obtained in this thesis and work done by previous researchers.

# Chapter 2

# Background Material

In this chapter several preliminaries and material that this thesis builds upon are introduced. This chapter is meant to refresh the readers about some of the principles and basics and does not constitute a reference aimed for fully understanding the presented material. Several references are cited along this chapter for further treatment of the topics introduced. This chapter first presents the difference between holonomic and nonholonomic constraints aiming to clarify the need to account for a robot's specific nonholonomic constraints in a motion planning algorithm. Next, the Dubins car model is introduced and its relevance to many systems that currently exist is presented. The modification made to the Dubins car model by many researchers in order to account for constant drifts is then explained briefly. After that basic results from optimal control theory are reviewed including Pontryagin's maximum principle along with an example to clarify the presented material. Finally, two point boundary value problems that usually emerge from the application of Pontryagin's maximum principle are discussed along with classical methods used to solve such problems.

## 2.1 Systems with Constraints

Generally, the freedom of motion of a mechanical system is constrained or restricted in certain ways. Constraints may be imposed on a system's position, velocity, etc. One specific method to categorize the constraints acting on the system classifies them as *holonomic* or *nonholonomic constraints*. Basic definitions of holonomic and nonholonomic constraints are presented in this section to refresh the readers' memroy. For a full treatment of mechanical constraints the reader may refer to Greenwood [36], Bloch [37] and Gans [38].

### 2.1.1 Holonomic Constraints

Consider a system described by $n$ generalized coordinates $q_1, q_2, ...., q_n$. *Holonomic constraints* are the constraints that can be written in the form of

$$f(q_1, ..., q_n, t) = 0. \tag{2.1}$$

These constraints depend exclusively on time and the generalized coordinates of a certain system. If holonomic constraints were to be written in terms of the time derivatives or velocities of the generalized coordinates $\dot{q}(t)$, then the equations resulting will emerge in a completely integrable form making it possible to integrate the system back to the form defined in equation (2.1). Holonomic constraints are also known as integrable constraints. A straightforward example of a system with holonomic constraints is the simple pendulum with a massless rigid rod. The fixed length of the massless rod that keeps the pendulum's mass at a fixed distance from its origin imposes a holonomic constraint on the system. Another example is an omni-wheeled robot. By combining different ratios of the rotational speed of the omni-wheels of the robot, an omni-wheeled robot is able to generate motion in any direction in the plane.

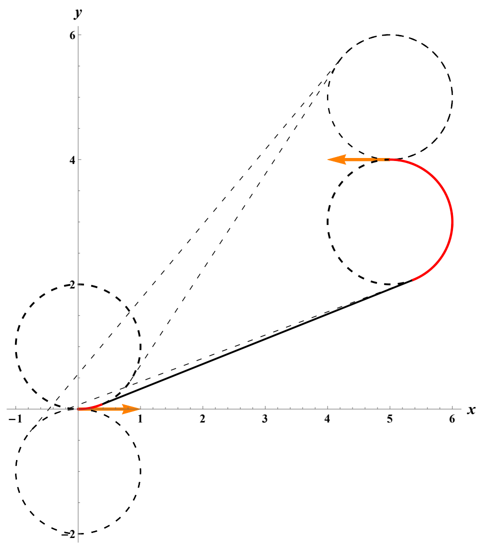
### 2.1.2 Nonholonomic Constraints

Nonholonomic constraints are constraints that cannot be written in terms of equation (2.1). Instead, nonholonomic constraints are written in terms of the differentials of the generalized coordinates of the system and time. The constraint equations take the form

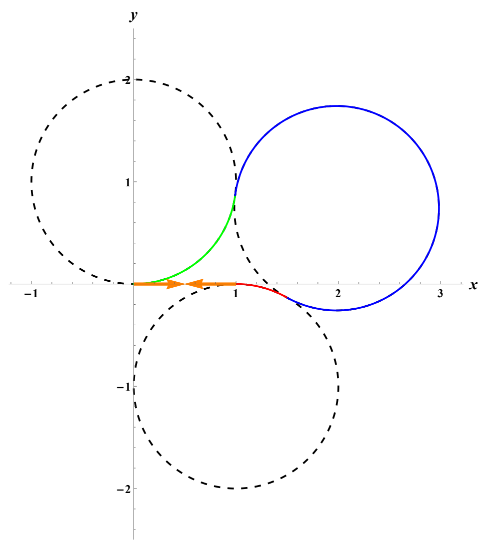
$$f(q_1, ..., q_n, \dot{q}_1, ..., \dot{q}_n, t) = 0, \tag{2.2}$$

where integrating the equations to retrieve equations that are similar in form to equation (2.1) is not possible. An example of a nonholonomic system is the rolling disk. A disk rolling on a horizontal surface has a constraint on its lateral motion. The lateral velocity of the rolling disk equals zero if the assumption of no skidding is made. This assumption imposes a nonholonomic constraint on the system. Other systems that posses nonholonomic constraints on their motion include cars, fixed wing airplanes, ships, certain underwater vehicles and many other systems.

Constraints imposed on the motion of a specific mechanical system restrict its motion in certain directions. This requires that motion planning algorithms developed for a certain system be customized towards maneuvers permitted by its constraints. The model of interest in this work describes the motion of various systems including, cars, fixed wing aircraft, ships and underwater vehicles. The model is known as the Dubins car model and it captures the constraint acting on the rotational velocity or turning rate of some systems or robots.




(a) Candidate Dubins Paths of Type CSC




(b) Candidate Dubins Paths of Type CCC

Figure 2.1: Dubins Car Possible Optimal Paths

## 2.2 Dubins' Paths

In the Dubins car model, Dubins [5], the system is modeled as a particle with a defined direction in a 2D plane. The path planning problem is to find the minimum time trajectory from an initial configuration to a final or target configuration. Let $q = (x, y, \theta)$ be an element of the configuration space $Q = \mathbb{R}^2 \times \mathbb{S}$. The kinematic model used in the formulation of the problem is presented in (2.3), where the particle is assumed to be moving at a constant forward velocity, $\nu$, which is typically equal to unity. Whereas $u(t) \in [-u_{max}, u_{max}]$ is the control input on steering with $u_{max} = 1$.

$$\begin{aligned} \dot{x}(t) &= \nu \cos\theta(t), \\ \dot{y}(t) &= \nu \sin\theta(t), \\ \dot{\theta}(t) &= u(t), \end{aligned} \tag{2.3}$$

The work done by Dubins [5] and Boissonnat [6] proves that candidate optimal paths consist of a maximum of *three* possible motion primitives. Namely maximum turning curves to either directions, left or right, denoted by $L$ and $R$ respectively, or a straight line denoted $S$. A total of *six* concatenations of motion primitives exist and take one of two forms, $CSC$ or $CCC$ where $C$ stands for either a right or a left curve. Paths of type $CCC$ may only exist if the distance between the initial and final configurations is less than four times the minimum turning radius. Both types of concatenations are presented in figure 2.1.

## 2.3 Model in Constant Currents

Another model proposed by McGee [14] in equation (2.4) introduces the translational effects that a constant flow will induce on the particle.

$$\begin{aligned} \dot{x}(t) &= \nu \cos\theta(t) + \eta \cos\phi, \\ \dot{y}(t) &= \nu \sin\theta(t) + \eta \sin\phi, \\ \dot{\theta}(t) &= u(t), \end{aligned} \tag{2.4}$$

where $\eta$ and $\phi$ are the speed and direction of the constant flow, respectively. Model (2.4), which shall be labeled as the *irrotational model*, was extensively used in the literature for solving the minimum time path planning problems for unmanned aerial vehicles and underwater autonomous vehicles. However, this model only accounts for the translational effects of the flow while it ignores any other interactions between the vehicle and the flow, namely, the rotational effects. From Techy [18], optimal paths for the *irrotational model* in eqn. 2.4 consist of maximum turning rate segments in either directions or straight lines. However, unlike Dubins' paths where a maximum turning rate curve produces a circle, a maximum turning rate curve for the *irrotational model* generates what is known as a *trochoid*. A *trochoid* is generated by a fixed point on a circle that rotates with its center translating along the flow direction. The concatenations of the motion primitives corresponding to the *irrotational model* is similar to that of the Dubins' paths.

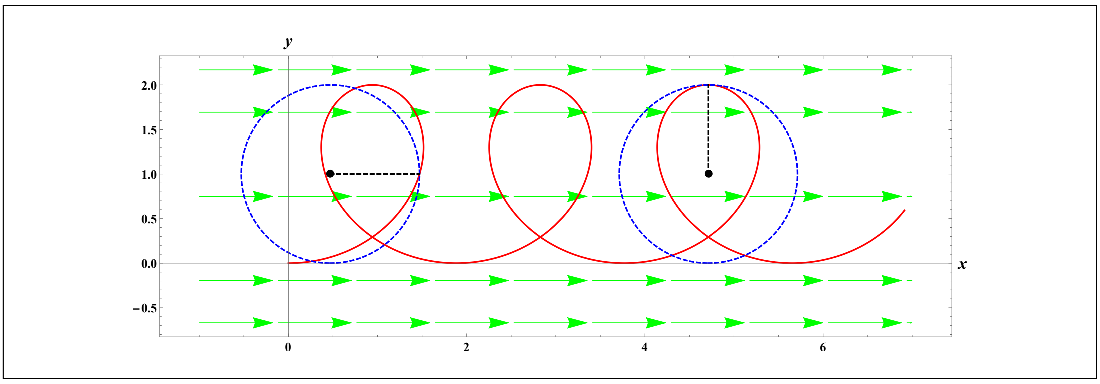



Figure 2.2: Trochoidal path due to a left maximum turning rate in constant flow

## 2.4 Optimal Control & Pontryagin's Minimum Principle

It is often required to find the control input for some system of the form

$$\dot{q}(t) = f\left(q\left(t\right), u\left(t\right), t\right) \tag{2.5}$$

such that the input control $u\left(t\right)$ minimizes a certain objective or cost and drives the system from a given initial state $q(t_0)$ to a final state $q(t_f)$. The branch in control theory that is concerned with finding such controls is known as optimal control theory. A typical optimal control problem includes a system of the form presented in eqn.(2.5) with the choice of an appropriate objective to optimize. The choice of such an objective is critical in defining the problem and finding an appropriate solution. This objective is known as a cost functional.

### 2.4.1 The Cost Functional

For an optimal control problem the general form of a cost functional is given by

$$J = h\left(q\left(t_f\right), t_f\right) + \int_{t_i}^{t_f} g\left(q\left(t\right), u\left(t\right), t\right) dt \tag{2.6}$$

where the first term $h\left(q\left(t_f\right), t_f\right)$ is known as the endpoint cost or the *Mayer cost*. The second term $\int_{t_i}^{t_f} g\left(q\left(t\right), u\left(t\right), t\right) dt$ is known as the running cost or the *Lagrange cost*. A functional maps a class of functions into an element in $\mathbb{R}$. Minimizing a functional is achieved using a branch in mathematics known as *Calculus of Variations*, Kirk [39]. For an optimal control problem, the cost functional is constrained by the system in eqn.(2.5) which makes the problem a constrained minimization problem.

### 2.4.2 Pontryagin's Minimum Principle

Using *Calculus of Variations,* Pontryagin [40], developed his principle known as *Pontryagin's Minimum, or Maximum Principle* ($PMP$). A deep discussion of PMP can be found in Kirk [39], Pontryagin [40], and Ross [41]. However, for the sole purpose of this work only a brief overview of the results obtained by Pontryagin will be presented without any derivations.

Given a system of the form

$$\dot{q}(t) = f\left(q\left(t\right), u\left(t\right), t\right) \tag{2.7}$$

associated with the cost function given by

$$J = h\left(q\left(t_f\right), t_f\right) + \int_{t_i}^{t_f} g\left(q\left(t\right), u\left(t\right), t\right) dt \tag{2.8}$$

Then it is possible to define the *control Hamiltonian* as

$$H = g\left(q\left(t\right), u\left(t\right), t\right) + \lambda^T . f\left(q\left(t\right), u\left(t\right), t\right) \tag{2.9}$$

where $\lambda$ is the known as the costate vector. An optimal control $u^*(t)$ is a control that globally minimizes the *Hamiltonian*. $q^*(t)$ and $\lambda^*(t)$ are the trajectory and costate vectors along which the *Hamiltonian* is minimum. Then the necessary conditions for $u^*(t)$ to be optimal as stated by Pontryagin for all $t \in \left[t_0, t_f\right]$ and all admissible $u(t)$ are as follows

$$\dot{q}^*(t) = \frac{\partial H}{\partial \lambda}\left(q^*(t), u^*(t), \lambda^*(t), t\right) \tag{2.10}$$

$$\dot{\lambda}^*(t) = -\frac{\partial H}{\partial q}\left(q^*(t), u^*(t), \lambda^*(t), t\right) \tag{2.11}$$

$$H\left(q^*(t), u^*(t), \lambda^*(t), t\right) \leqslant H\left(q^*(t), u(t), \lambda^*(t), t\right) \tag{2.12}$$

$$\begin{aligned}[\frac{\partial h}{\partial q}\left(q^*(t_f), t_f\right) - \lambda^*\left(t_f\right)]^T \delta q_f \\ + \left[H\left(q^*(t_f), u^*(t_f), \lambda^*(t_f), t_f\right) + \frac{\partial h}{\partial t}\left(q^*(t_f), t_f\right)\right] \delta t_f = 0\end{aligned} \tag{2.13}$$

Where $\delta q_f$ and $\delta t_f$ are the first variation in time and state. If the problem is a fixed final state problem then $\delta q_f$ is zero, so is $\delta t_f$ when the problem is fixed end time problem. This summarizes the necessary conditions an optimal control should satisfy. However, it is very important to point out that PMP doesn't guarantee the existence of an optimal control and doesn't provide an exact analytic form for the control. Thus it is possible to find controls satisfying the PMP conditions yet not minimizing or maximizing the cost functional.

To find $u^*(t)$ where $H$ is minimum, it is possible to compute the first and second derivatives of the *Hamiltonian*.

$$\frac{\partial H}{\partial u}\left(q^*(t), u^*(t), \lambda^*(t), t\right) = 0 \tag{2.14}$$

$$\frac{\partial^2 H}{\partial u^2}\left(q^*(t), u^*(t), \lambda^*(t), t\right) = 0 \tag{2.15}$$

If the control is not bounded then setting the first derivative to zero and solving for $u^*(t)$ might yield an analytic form for the optimal control in very few cases. The second derivative has to be a positive definite matrix to guarantee local optimality (the *Hamiltonian* being a local minimum).

### 2.4.3 Example: The Double Integrator

To find the optimal control $u^*(t)$ that transfers the double integrator system given by

$$f\left(q(t), u(t), t\right) = \begin{cases} \dot{q_1}(t) & = q_2(t) \\ \dot{q_2}(t) & = u(t) \end{cases} \tag{2.16}$$

from any initial state $q(t_0)$ to the origin in minimum time while satisfying

$$|u(t)| \leqslant 1. \tag{2.17}$$

First the control Hamiltonian is constructed and the control input $u(t)$ is studied to minimize $H$.

$$H = 1 + \lambda_1(t)x_2(t) + \lambda_2(t)u(t) \tag{2.18}$$

It is clear that the hamiltonian $H$ is linear in control. Kirk [39] provides a proof that a singular interval cannot exist for this type of problem, thus $\lambda_2(t) \neq 0$ for all $t \in \left[t_0, t_f\right]$. Hence, in order to minimize $\mathbb{H}$ the optimal control $u^*(t)$ is given by

$$u^*(t) = \begin{cases} -1 & \text{if } \lambda_2(t) > 0 \\ 1 & \text{if } \lambda_2(t) < 0 \end{cases} \tag{2.19}$$

From eqn. (2.11) the costates are given by

$$\dot{\lambda}^*(t) = \begin{cases} \dot{\lambda}_1^*(t) = 0 \\ \dot{\lambda}_2^*(t) = -\lambda_1^*(t). \end{cases} \tag{2.20}$$

Integrating both equations yields an analytical solution for the optimal costates which are presented in terms of the constants of integration $c_1$ and $c_2$

$$\lambda^*(t) = \begin{cases} \lambda_1^*(t) = c_1 \\ \lambda_2^*(t) = -c_1 t + c_2. \end{cases} \tag{2.21}$$

The solution for $\lambda_2^*(t)$ in eqn. (2.21) shows that there can be *at most* one switching of the control input. The solution for the optimal trajectory of the system is given by eq. (2.10) as

$$f^*(t) = \begin{cases} \dot{q}_1^*(t) & = q_2^*(t) \\ \dot{q}_2^*(t) & = u^*(t). \end{cases} \tag{2.22}$$

The analytical solution for the optimal trajectory of the system can be obtained by integrating eqns. (2.22) for both cases of the optimal control input $u^*(t) = \pm 1$. The solution is then given in terms of the constants of integration $c_3$ and $c_4$ as follows,

$$q^*(t) = \begin{cases} q_1^*(t) & = \pm \frac{1}{2} t^2 + c_3 t + c_4 \\ q_2^*(t) & = \pm t + c_3. \end{cases} \tag{2.23}$$

In order to find the constants of integration $c_1$, $c_2$, $c_3$, and $c_4$ the initial and final conditions for the problem must be incorporated in the solution.

However, it is not always possible to find closed form solutions for optimal control problems as in the example of the double integrator. For complicated nonlinear systems the resulting optimal trajectory and costate dynamics may come out as nonlinear differential equations making the problem harder to solve. Since closed form solutions for nonlinear differential equations cease to exist in most cases, numerical and discretization methods are used instead to arrive at approximate solutions for the presented problem. For a rigorous discussion of methods used to solve such problems the reader may refer to Kirk [39], Lewis [42], and bertsekas [43] and [44].

## 2.5 Two Point Boundary Value Problems

In a standard path planning problem it is required to find a trajectory $q(t)$, for a certain system $\dot{q(t)} = f\left(q, u, t\right)$, that connects its initial configuration $q\left(t_0\right)$ with a desired final configuration $q\left(t_f\right)$. When using optimal control techniques, particularly PMP, the system dynamics $f\left(q, u, t\right)$ are projected onto a control hamiltonian $H$, the costate dynamics are derived, and the optimal controls are expressed in terms of the states and costates of the system. This leads to a system of $2n$ differential equations composed of $n$ state dynamics equations and $n$ costate dynamics equations. However, The boundary conditions are provided on the initial and final states of the system and no prior knowledge is provided regarding the costates other than their dynamics. A brief overview of some methods used to solve *Two Point Boundary Value Problems* is provided in this section. For a thorough explanation and presentation of the methods used to solve two point boundary value problems the reader can refer to Ward [45] or Press [46].

### 2.5.1 Shooting Method

Boundary value problems are considerably more difficult than initial value problems. A shooting algorithm starts with a guess on the remaining needed *initial conditions* then integrates the system forward in time until it reaches the desired final time $t_f$. Then the final values of the states are checked, it the actual final states match the target states or the final conditions are met then the algorithm has converged. Else the initial guess is modified and the process is repeated until the algorithm converges. Interpolation is mostly used to improve the guess. The shooting method is time consuming and is not always guaranteed to converge particularly for nonlinear multi-dimensional systems.

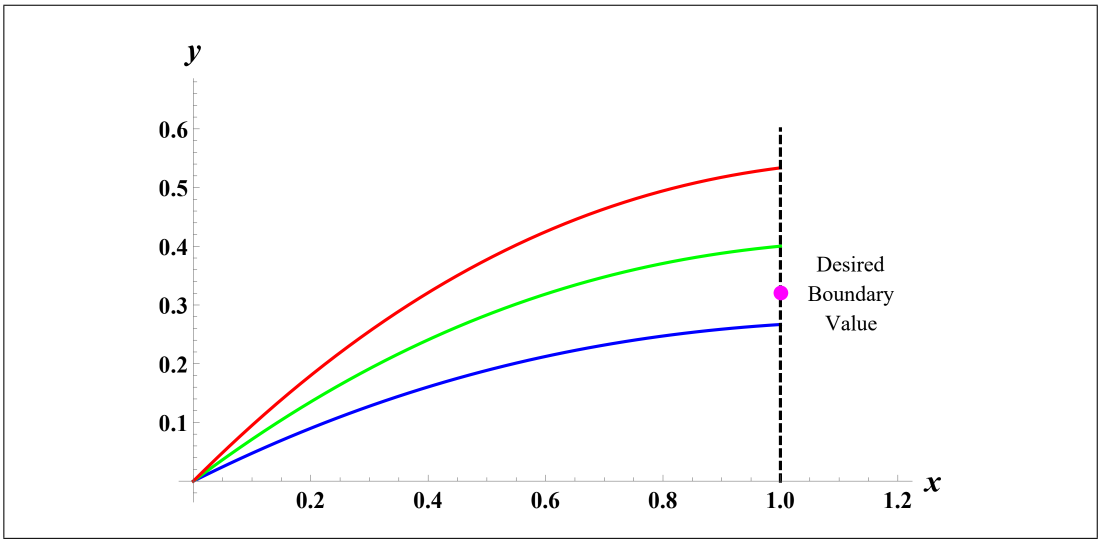



Figure 2.3: Shooting Method for a 1-Dimensional Differential Equation

### 2.5.2 Relaxation Method

In relaxation methods, instead of converting the boundary value problem into an initial value problem, the ODEs are replaced by *finite difference equations* on a mesh that spans the independent variable domain. The relaxation method then starts with an initial guess of the solution over the entire independent variable domain as shown in figure 2.4, then improves it using available iterative schemes such as the multi-dimensional Newton's method.

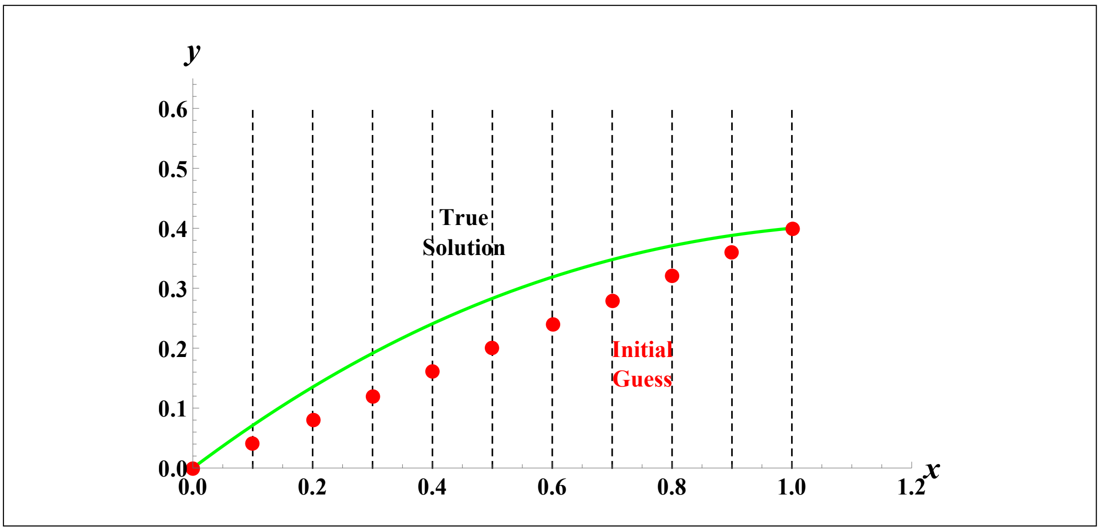



Figure 2.4: Relaxation Method for a 1-Dimensional ODE with an Initial Guess

The methods presented above are not the only methods available for solving *two point boundary value problems*. Over the years many methods emerged in attempt to solve such problems. Most methods rely on numerical approximations and iterative schemes in order to provide approximate solutions for two point boundary value problems. Full a full treatment of boundary value problems a curious reader me refer to Gakhov [47].

# Chapter 3

# New Kinematic Model in Constant Currents

This chapter presents the new model developed to describe the motion of vehicles having similar constraints to that of a Dubins car model while traveling in mediums with constant flow fields. First the motivation behind the need for a new model is discussed. Then the model derivation is presented. Finally, simulation results are presented to compare the model performance with previously developed models.

## 3.1 Motivation

The Dubins car model describes the constraint of a minimum turning radius that many vehicles or robot posses. A modified model used extensively in the literature, [14], [16], [17], [18], and [35] describes how a constant flow will affect the translation components of the Dubins car model. However, the modified model considers the vehicle as a massless particle that doesn't interact with its environment. The dynamics of an autonomous underwater vehicle AUV or an airplane moving in flow field are much more complicated. The flow field, and due to the geometry of the traveling vehicle, will cause it to rotate and not only translate. Thus there is a need for developing a simplified kinematic model that is still easy to work with for path planning purposes, but models more accurately how a vehicle may react to external constant flows. The model developed in this chapter is inspired by the hybrid autonomous underwater vehicle being currently developed at the *Vision and Robotics Lab at AUB*. The vehicle is shown in figure 1.2 and the flow is assumed to rotate the vehicle until its axis of symmetry is aligned with the flow direction.

## 3.2 Model Derivation

In order to capture more details of how an interaction might cause the vehicle to rotate, an additional term is added to the third equation in (2.4) to arrive at the novel model given in eqn. (3.1). This proposed model, labeled as the *rotational model*, assumes that the effect on rate of change of the heading angle is proportional to the perpendicular component of the flow velocity with respect to the heading angle of the vehicle as shown in Fig. 3.1.

$$\begin{aligned} \dot{x}(t) &= \nu(t)\cos\theta(t) + \eta\cos\phi, \\ \dot{y}(t) &= \nu(t)\sin\theta(t) + \eta\sin\phi, \\ \dot{\theta}(t) &= u(t) + \rho\eta\sin\big(\phi - \theta(t)\big), \end{aligned} \tag{3.1}$$

where the proportional constant, $\rho$, captures a geometric attribute of the body of the vehicle, the location of the center of mass, and the nature of the interaction between the vehicle and the flow medium.

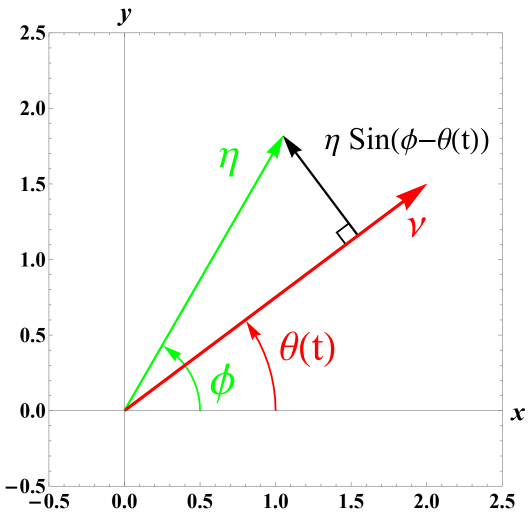



Figure 3.1: Vehicle's forward speed $\nu$, flow speed $\eta$ and their corresponding direction angles $\theta$ and $\phi$

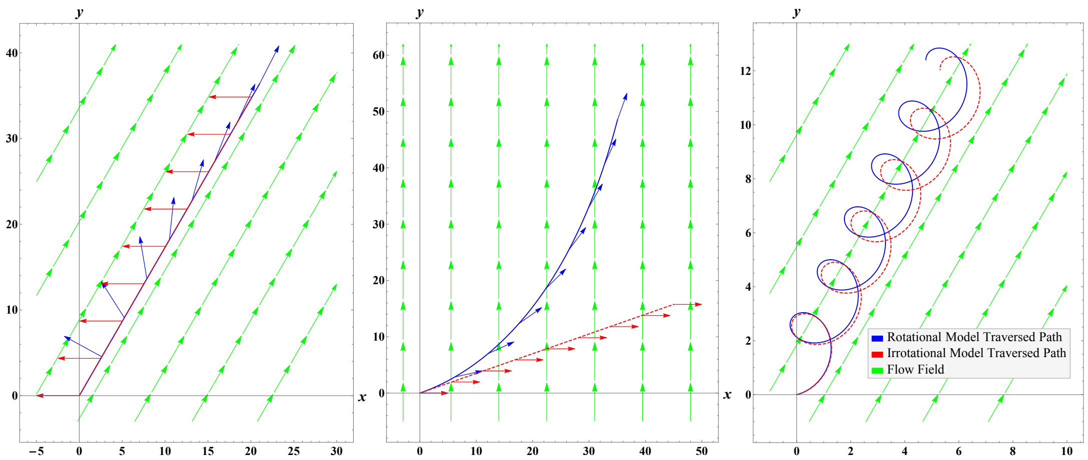
To clarify the significance of the developed rotational model (3.1), several simulations were worked out and the motion results were compared to the irrotational model in (2.4). For a given planar curve $(x(t), y(t))$, let $\psi = tan^{-1}\big(\dot{x}(t), \dot{y}(t)\big)$ be the course angle representing the angle that the tangent to the curve makes with the horizontal axis.

## 3.3 Comparison with Previous Models

**Case 1:** $u = 0$ and forward velocity $\nu = 0$. This case corresponds to placing the AUV in a constant current with a zero steering control $u = 0$ and a zero forward velocity $\nu = 0$. In this case the constant flow is the only effect acting on the vehicle, the motion of the vehicle as described by both the irrotational (2.4) and rotational (3.1) models is shown in Fig.3.2a.

It is clear that the heading angle for model (2.4) stays constant where as the heading angle of model (3.1) $\theta(t) \to \phi$ as $t \to \infty$. That is the vehicle will eventually align with the flow. However, The course angle $\psi(t) = \phi$ for both models stays constant.

**Case 2:** $u = 0$ and forward velocity $\nu = 1$. In this case the forward velocity is assumed be maximum at unity while the steering control is kept at zero. The resulting traversed paths are shown in Fig.3.2b. With no steering input applied to either models, $\psi(t)$ corresponding to the irrotational model (2.4) is constant and experiences no change, while $\psi(t)$ corresponding to the rotational model (3.1) is eventually aligned with $\phi$ and $\theta(t)$ as $t \to \infty$.




(a) Case 1 with $u = 0$, $\nu = 0$, $\theta_{init} = \pi$, $\rho = 0.1$, $\eta = 0.35$, and $\phi = \frac{\pi}{3}$

(b) Case 2 with $u = 0$, $\nu = 1$, $\theta_{init} = 0$, $\rho = 0.1$, $\eta = 0.35$, and $\phi = \frac{\pi}{2}$

(c) Case 3 with $u = 1$, $\nu = 1$, $\theta_{init} = 0$, $\rho = 0.1$, $\eta = 0.35$, and $\phi = \frac{\pi}{3}$

Figure 3.2: Rotational model vs Irrotational model for various control inputs

**Case 3:** $u = 1$ and forward velocity $\nu = 1$. The model in (2.4) traverses a trochoidal path as described in [17] and [18]. However, the developed rotational model (3.1) shows a variation from the results described by (2.4) as shown in Fig.3.2c. It is important to stress that $\theta(t)$ is not necessarily equal to the course angle $\psi(t)$ for either models.

From the above simulations, it is obvious that the proposed model better captures the motion of a real system moving in a medium with a constant drift field.

# Chapter 4

# Minimum Time Trajectories

This chapter studies the model developed in Chapter 3. It aims to arrive at a solution describing the trajectories corresponding to the minimum time of travel for the model in eqn. (3.1) while traveling in environments with constant flow fields. First, with the help of optimal control theory, discrete control values that the optimal time trajectory consist of are derived. These discrete optimal control values are then used to describe fixed motion primitives, or geometrically defined shapes that build up the minimum time trajectories. After that the work done by Dolinskaya [21] is used to arrive at a concatenation of the motion primitives that guarantees a minimum trajectory. Next, a condition is imposed on the flow speed to guarantee that the optimal trajectory is one of two possible concatenation families possible. Then, two algorithms are introduced to solve for the two possible families of optimal trajectories and find the optimal path. Finally simulations are presented to show the resulting time optimal trajectories resulting compared to time optimal trajectories of the model that doesn't account for the rotational effect.

## 4.1 Time Optimal Controls

In order to solve the path planning problem, we take recourse to optimal control theory techniques, specifically, Pontryagin's Minimum Principle (PMP). PMP is used to reduce the space of candidate optimal paths. For a minimum time problem where the end-time $t_f$ is free, the cost function is given by

$$J = \int_0^{t_f} dt = t_f, \tag{4.1}$$

and the control Hamiltonian is given by

$$H(\lambda, q, u, t) = g(q, u, t) + \lambda^T f(q, u, t), \tag{4.2}$$

where $g$ is the integrand of the cost function $J$ and $f : (q, u, t) \mapsto \dot{q}$ is the kinematic model given in (3.1). Using the proposed model, the Hamiltonian becomes

$$\begin{aligned} H(\lambda, q, u, t) =& 1 + \lambda_x(\cos\theta(t) + \eta\cos\phi) + \\ & \lambda_y(\sin\theta(t) + \eta\sin\phi) + \\ & \lambda_\theta(u(t) + \rho\eta\sin\big(\phi - \theta(t)\big)), \end{aligned} \tag{4.3}$$

where $\lambda = (\lambda_x, \lambda_y, \lambda_\theta)$ is the vector of co-state variables. These variables are governed by the costate equations given by

$$\dot{\lambda}_x = -\frac{\partial H}{\partial x} = 0, \tag{4.4}$$

$$\dot{\lambda}_y = -\frac{\partial H}{\partial y} = 0, \tag{4.5}$$

$$\begin{aligned} \dot{\lambda}_\theta &= -\frac{\partial H}{\partial \theta} \\ &= (\lambda_x - \lambda_\theta\rho\eta\cos\phi)\sin\theta(t) \\ &\quad + (\lambda_\theta\rho\eta\sin\phi - \lambda_y)\cos\theta(t). \end{aligned} \tag{4.6}$$

According to Pontryagin's Minimum Principle the optimal control, $u^*$, is the one that minimizes the Hamiltonian such that, $H(\lambda^*, q^*, u^*, t) \leqslant H(\lambda^*, q^*, u, t)$. Also for a free end-time problem, using [39], the boundary condition is given by $H(\lambda^*(t_f), q^*(t_f), u^*(t_f), t_f) = 0$. Additionally, if the Hamiltonian is not explicit in time then we get

$$H(\lambda^*(t), q^*(t), u^*(t)) = 0. \tag{4.7}$$

For the proposed model and its associated Hamioltonian in (4.3), the control $u$ is only multiplied by the costate $\lambda_\theta$. Thus, minimizing the Hamiltonian with respect to $u(t)$ is equivalent to minimizing $\lambda_\theta(t)u(t)$. Thus, the optimal control, $u^*(t)$, is a function of the sign of $\lambda_\theta(t)$ which is also known as the *switching* function. To find the optimal controls for the proposed model the following cases are considered.

### 4.1.1 Case where $\lambda_\theta = 0$

Using (4.7) the Hamiltonian becomes $H = 1+\lambda_x(\cos\theta(t)+\eta\cos\phi)+\lambda_y(\sin\theta(t)+\eta\sin\phi) = 0$. Even though $u(t)$ doesn't appear explicitly, it is still possible to deduce important results regarding the control input. From the costate equations, (4.4) and (4.5), and from the assumption of constant flow, the following variables, $\lambda_x$, $\lambda_y$, $\eta$, and $\phi$, are constants. Thus, the Hamiltonian becomes

$$\lambda_x\cos\theta(t) + \lambda_y\sin\theta(t) = \textit{constant}, \tag{4.8}$$

which implies that $\theta(t) = \textit{constant} \pmod{2\pi}$. Thus, the heading angle, $\psi(t)$, is constant which yields a straight line path. Additionally, since $\theta(t)$ is constant, using the third equation in (3.1), one can solve for the optimal control, which will be referred to as a singular control, to get

$$u^*(t) = -\rho\eta \sin\left(\phi - \theta(t)\right). \tag{4.9}$$

### 4.1.2 Case where $\lambda_\theta \neq 0$

The optimal control input, $u^*(t)$, that minimizes the Hamiltonian whenever $\lambda_\theta \neq 0$ is given by,

$$u^* = -sign(\lambda_\theta)u_{max}, \tag{4.10}$$

which represents a maximum turning control either to the left or the right, also known as a maximum effort or a bang control given by

$$u^*(t) = \pm 1. \tag{4.11}$$

Hence, Pontryagin's Minimum Principle reduces the space of candidate minimum time paths to those traversed by the model under the two optimal controls, the singular controls in (4.9) resulting in a straight line paths, and the maximum effort controls in (4.11) resulting in stretched-like trochoids.

## 4.2 Structure of Motion Primitives

Having reduced the space of candidate minimum time paths to concatenations of straight lines and turning segments corresponding to maximum effort control inputs, in this section, these motion primitives are characterized and their structure is analyzed to be used in the path synthesis problem. First some terminology is introduced.

- $C_a$ : A curve traversed by the model when taking a maximum turn rate either to the left or the the right so $C$ could take the values of $L$ or $R$ and $a$ is either $i$ indicating *initial* or $f$ indicating $final$.

- $t_{iL}$ : time spent traversing a maximum left turning segment starting from the initial configuration $q_i = \{x_i, y_i, \theta_i\}$. Similarly $t_{iR}$ could be defined to correspond to a right turn.

- $t_{fL}$ : time spent traversing a maximum left turning segment ending at the final configuration $q_f = \left\{x_f, y_f, \theta_f\right\}$. Similarly $t_{fR}$ could be defined to correspond to a right turn.

- $q_{iL}(t)$ : corresponds to the configuration of a point starting at an initial configuration and traversing a maximum left turn for a duration $t$. Similarly $q_{iR}(t)$ is defined for a right turn.

- $S_{C_i,C_f}$ : is the straight line segment that is tangent to both the initial turning segment $C_i$ and final turning segment $C_f$.
- $V_{C_i,C_f}$ : velocity of the particle while traveling along the straight segment tangent to both $C_i$ and $C_f$.

### 4.2.1 Maximum Turning Rate Segments

Pontryagin's Minimum Principle restricts the segments of any optimal path for the rotational model (3.1) into maximum turning rate segments and straight line segments. Some important implications of such results are presented below.

Using Mathematica, an analytical solution for the non-linear differential equation describing $\theta(t)$ in the rotational model in (3.1) is given by

$$\theta(t) = \phi + 2\tan^{-1}\left(\frac{\eta\rho + \sqrt{u^2 - \eta^2\rho^2}\tan\beta)}{u}\right) \tag{4.12}$$

where $\beta$ is

$$\beta = \frac{1}{2}t\sqrt{u^2 - \eta^2\rho^2} + \tan^{-1}\left(\frac{u\tan\left(\frac{\theta_0 - \phi}{2}\right) - \eta\rho}{\sqrt{u^2 - \eta^2\rho^2}}\right). \tag{4.13}$$

Inspecting the above solution, specifically the term associated with time, $t$, it is clear that $\theta(t)$ is periodic with a period

$$P = \frac{2\pi}{\sqrt{u^2 - \rho^2\eta^2}} \tag{4.14}$$

Under constant flow conditions and a constant bang control, such as $u(t) = 1$, the net displacement vector generated by a full turning period of $\theta(t)$, denoted by $D_{2\pi}$, is *unique* and independent of the initial configuration. That is, starting from any initial fixed position $\{x(0), y(0)\}$, then taking a maximum rate left turn for a period $P$, the final position $\{x(P), y(P)\}$ is the same independent of the initial heading $\theta(0)$ as shown in Fig. 4.1. The arrows in Fig. 4.1 depict the initial and final headings corresponding to each maximum left turn curve. Similarly $u(t) = -1$ defines a unique net displacement vector for a full period right turn. Hence, the shapes of bang path are *periodic*. This could be verified by integrating the first two equations in (3.1) over a full period of $\theta(t)$.

### 4.2.2 Straight Segments

For the case where $\theta$ is constant, and given that $u \in [-1, 1]$, it is clear from eqn. (4.8) that to achieve singular segments, one must assume that $|\rho\eta| < 1$ .

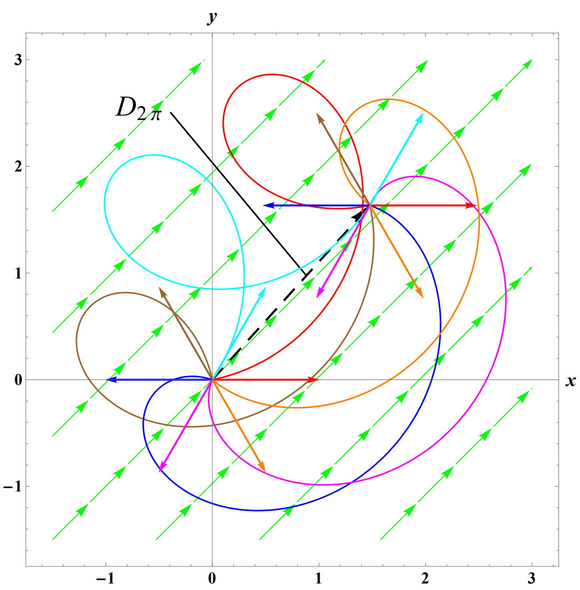



Figure 4.1: Full Period Maximum Effort Left Turns starting form different initial headings under fixed flow conditions result in a fixed displacement vector

Unlike prior models where the control $u$ takes only the discrete values $1, 0$, or $-1$, in the proposed model, for singular segments, $u$ could take any feasible value to ensure that $\theta$ is constant. Thus, for the rest of this work, it is assumed that flow parameters are constrained by $|\rho\eta| < 1$.

## 4.3 Optimal Path Synthesis

Having proved that any time optimal path must consist of either singular straight segments or bang maximum turning rate segments, the number and order of concatenations of such segments, that is, the path synthesis problem, is addressed next.

### 4.3.1 Concatenations of Possible Path Segments

The first two equations of the proposed rotational model (3.1) describing $\dot{x}(t)$ and $\dot{y}(t)$ can be written in the form of the model introduced by [22] which is given by

$$
\begin{aligned}
\dot{x}(t) &= V(\theta)\cos\theta(t), \\
\dot{y}(t) &= V(\theta)\sin\theta(t), \\
\dot{\theta}(t) &= \frac{V(\theta)}{R(\theta)}u(t),
\end{aligned}
\tag{4.15}
$$

where $V(\theta)$ is the vehicle's net speed in polar coordinates and $R(\theta)$ is the minimum turning radius of the vehicle also in polar coordinates. Recall that the two components, $x(t)$ and $y(t)$, describe the path traversed by the vehicle. Thus, results obtained by [22] on path segment concatenations are applicable to the proposed rotational model in (3.1). Hence, as indicated in [22], the number of segment concatenations and the structure of the path are a function of the convexity of $V(\theta)$, and can be summarized as follows:

### 4.3.2 Non-Convex Speed Polar Plots

If the line connecting the initial and final configurations intersects a non-convex portion of the speed polar plot, the optimal path is comprised of *five* segments having the following specific order $CSCSC$ where the straight lines are parallel to the straight lines to both edges of the convex hull of the non-convex region as shown below in fig. 4.2.

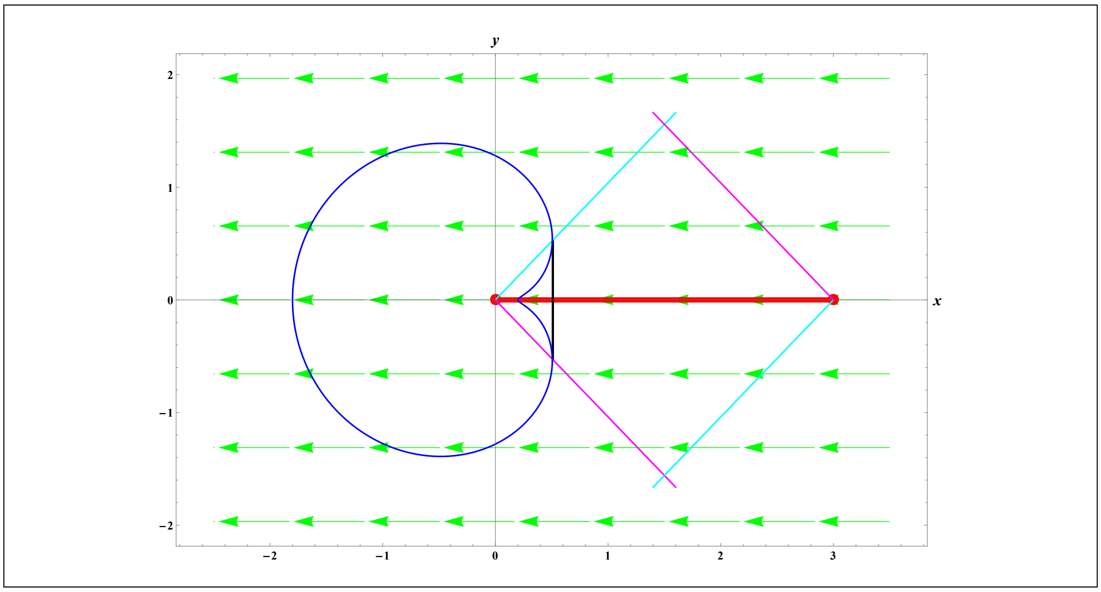



Figure 4.2: Speed Polar Plot with Non-convex Region and a Convex Hull

### 4.3.3 Convex Speed Polar Plots

If the line connecting the initial and final configurations intersects a convex portion of the speed polar plot, then the optimal path is comprised of at most *three* sections which have the following order $CSC$ or $CCC$.

## 4.4 Convexity Condition

Restricting the path synthesis problem in this paper to either $CSC$ to $CCC$, another constraint is imposed on the flow parameters in such a way to ensure that the speed polar plot is convex everywhere.

The speed in polar coordinates for the proposed model is given by

$$V(\theta) = \sqrt{\dot{x}^2 + \dot{y}^2} = \sqrt{\eta^2 + 2\eta\cos(\theta - \phi) + 1}.$$

Note that, the net speed polar plot in the absence of any flow, *i.e.* $\eta = 0$, is simply the unit circle depicted as a dashed circle in Fig.4.3. The effect of constant flow will deform this unit circle, where the amount and location of deformation, as expected, depends on the flow parameters, $\eta$ and $\phi$.

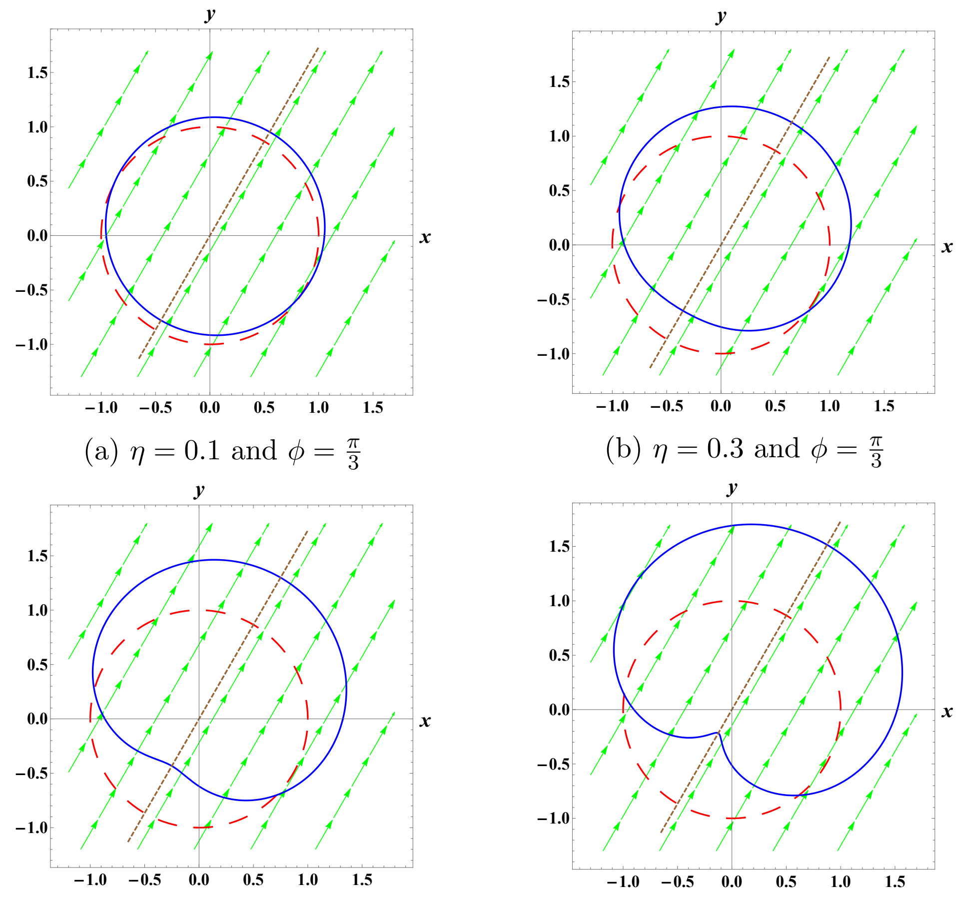
Studying the effect of the constant flow on the speed polar plot, as shown in Fig. 4.3, it is clear that the maximum deformation to the polar plot will occur along the direction of the flow, that is, at $\theta = \phi + \pi$. Let $\alpha = \theta - \phi$, then the curvature in polar coordinates is given by

$$\kappa = \frac{V^2 + 2\left(\frac{\partial V}{\partial \alpha}\right)^2 - V\frac{\partial^2 V}{\partial \alpha\,\partial \alpha}}{\left(V^2 + \left(\frac{\partial V}{\partial \alpha}\right)^2\right)^{3/2}}. \tag{4.16}$$

Setting the numerator of the curvature, $\kappa$, in (4.16) to zero, will determine the flow speed, $\eta$, at which the deformation in the speed polar plot starts to form a non-convex region in the plot. The numerator in terms of $\alpha$ is given by

$$\begin{aligned} 10\left(\eta^3 + \eta\right)\cos(\alpha) + 3\eta^2\cos(2\alpha) \\ +2\eta^4 + 13\eta^2 + 2 = 0. \end{aligned} \tag{4.17}$$

Setting $\alpha = \pi$, then the only feasible root of the equation turns out to be $\eta = \frac{1}{2}\left(3 - \sqrt{5}\right)$ which is about 38.2% of the vehicle's maximum forward speed and is the maximum flow velocity for which the entire speed polar plot will remain convex.



(a) $\eta = 0.1$ and $\phi = \frac{\pi}{3}$

(b) $\eta = 0.3$ and $\phi = \frac{\pi}{3}$

(c) $\eta = 0.5$ and $\phi = \frac{\pi}{3}$

(d) $\eta = 0.75$ and $\phi = \frac{\pi}{3}$

Figure 4.3: Polar plots of the maximum net speed for different $\eta$

Having characterized the individual segments and the concatenations of such segments, and by setting $\eta \leq 0.382$, the model is then restricted to the case where speed polar plot is convex. This restriction guarantees that an optimal path takes either the form $CSC$ or $CCC$. This reduces the space of possible optimal paths to a maximum of six types: $LSL$, $RSR$, $RSL$, $LSR$, $LRL$, and $RLR$.

## 4.5 CSC Type Path Synthesis

There exists four candidate minimum time paths of type $CSC$ between an initial configuration and a final one, namely $LSL$, $LSR$, $RSR$, and $RSL$. For an initial and final configurations where the Euclidean distance between $q_i$ and $q_f$ is greater than $4\left|\left|D_{2\pi}\right|\right|$ it is guaranteed that the solution will be of $CSC$ type.

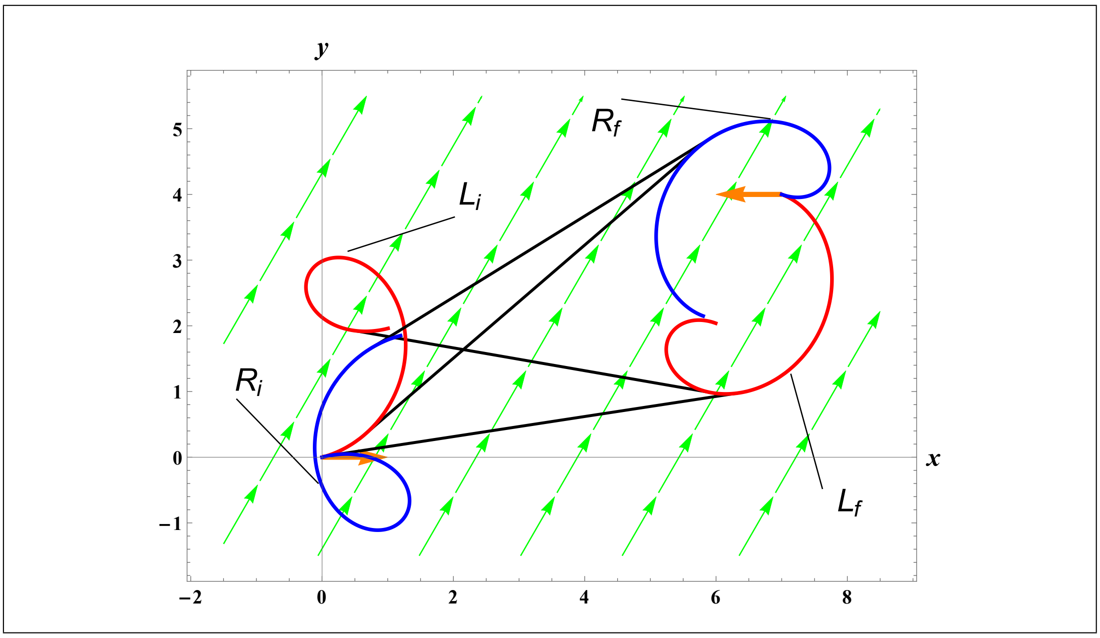



Figure 4.4: $CSC$ path synthesis for $q_f = \{8, 4, \pi\}$, $\eta = 0.35$, and $\phi = \frac{\pi}{3}$

Algorithm 1 constructs all four possible $CSC$ paths then finds the path corresponding to the minimum travel time. In order to accomplish this, the algorithm first constructs the initial maximum turning rate curves starting from $q_i = \{x(0), y(0), \theta(0)\}$ to the right and left respectively labeled as $R_i$ and $L_i$. Next, maximum turning rate segments are constructed to end at $q_f = \{x(P), y(P), \theta(P)\}$ turning to the right and left respectively labeled as $R_f$ and $L_f$. Then, the algorithm seeks a line segment connecting a point on $C_i$ to a point on $C_f$ such that the line segment is tangent to both curves. Finally, the algorithm computes the total time of travel along each of the four possible paths and finds the path corresponding to the minimum travel time. Fig.4.4 shows the initial and final maximum rate turning curves for a full period $P$ along with the corresponding tangents.

## 4.6 CCC Type Path Synthesis

For cases where the Euclidean distance between $q_i$ and $q_f$ is less than $4\,||D_{2\pi}||$, a mid-curve or type $C_m$ could possibly be tangent to both $C_i$ and $C_f$. Fig.4.5 shows an example where multiple right turning middle curves, depicted by dashed curves, starting initially from different points along the initial left turning segment intersect with the final turning segment. Also note that one of the mid-curves,

**Algorithm 1** *CSC Type Synthesis*

**Given** $q_i$, $q_f$, $\eta$, $\phi$, and $\rho$
**Simulate** $f(q,u,t)$ for $t \in [0,P]$ **with boundary condition** $f(q,\pm 1,0) = q_i$ to get $C_i$
**Simulate** $f(q,u,t)$ for $t \in [0,P]$ **with boundary condition** $f(q,\pm 1,P) = q_f$ to get $C_f$

---

**for** The 4 possible combinations of $C_i$ and $C_f$ **do**
  **Solve** for $t_{i,C_i,C_f}$ **and** $t_{f,C_i,C_f}$
  $\tan^{-1}\left[\left(x_{C_f} - x_{C_i}\right), \left(y_{C_f} - y_{C_i}\right)\right] = \tan^{-1}\left[\dot{x}_{C_i}, \dot{y}_{C_i}\right]$
  $\tan^{-1}\left[\left(x_{C_f} - x_{C_i}\right), \left(y_{C_f} - y_{C_i}\right)\right] = \tan^{-1}\left[\dot{x}_{C_f}, \dot{y}_{C_f}\right]$
  **Compute** $t_{sC_iC_f} = \left\|S_{C_i,C_f}\right\| / V_{C_iC_f}$
  **Compute** $t_{fC_iC_f} = P - t_{fC_iC_f}$
  **Compute** $T_{C_iC_f} = t_{iC_iC_f} + t_{sC_iC_f} + t_{fC_iC_f}$
**end for**
**FindMin** $T_{C_iC_f}$

depcited as a solid curve, is also tangent to $L_f$. For such case, a $CCC$ type curve is a possible minimum time candidate. Algorithm 2 solves for $CCC$ candidates only. Possible path types are either $RLR$ or $LRL$.

Algorithm 2 first constructs both $L$ and $R$ initial and final turning segments. After that a turning segment with opposite direction to that of the initial turning segment and having initial conditions be any point belonging to the initial turning segment is constructed. Then for each pair of same direction initial and final turning segments $C_i$ and $C_f$, an opposite direction turning segment $C_m$ is solved for numerically. $C_m$ starts from a point and a heading on $C_i$ and is tangent to $C_f$. Finally the total travel time corresponding to each candidate path is computed and and the path corresponding to the minimum travel time is found. However, for the case where $CCC$ paths are possible candidates, $CSC$ paths are also still valid candidates for minimum time paths. Thus, an algorithm combining both algorithms 1 and 2 computes the minimum time path in such a case.

## 4.7 Simulations

Some results of the algorithms introduced earlier are presented in this section. Without loss of generality, the initial configuration is assumed to be $q_i = \{0, 0, 0\}$ for all cases. Also, $\rho$ is assumed to equal 0.1 for all cases presented in Fig. 4.6

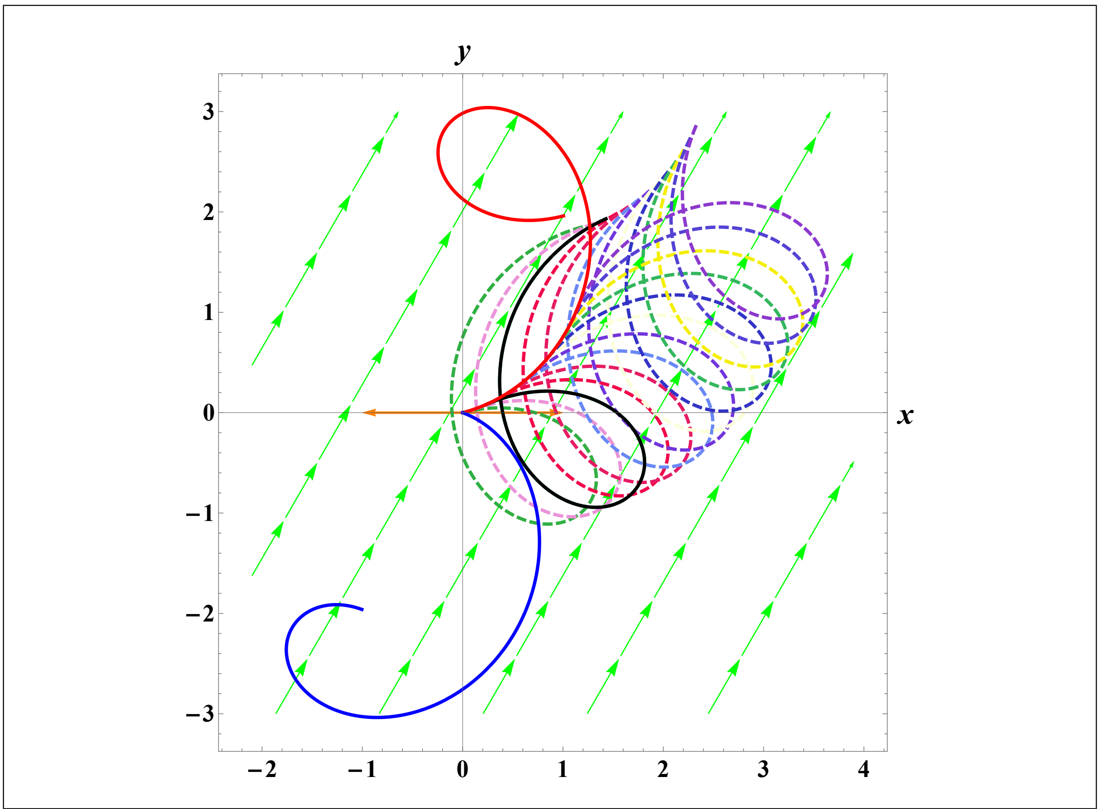



Figure 4.5: $L_i$ and $L_f$ with $q_i = (0, 0, 0)$, $q_f = (0, 0, \pi)$ and multiple right turning segments starting on $L_i$

and Fig. 4.7. The flow velocity $\eta$ is kept close to the limit, that is 0.38, to ensure that the speed polar plot is convex.

Fig. 4.6 depicts cases where all possible path concatenations are of the type $CSC$. Three different cases are presented where the final configuration, $q_f$, as well as the flow parameters, $\phi$ and $\eta$, are changed. For each case, the four possible path concatenations $LSL$, $RSR$, $LSR$ and $RSL$ are always guaranteed to exist as depicted by the dashed curves. The path corresponding to the minimum time of travel of the vehicle is depicted using solid curves. Note that, the turning radius and length of various turning segments depend on the flow parameters as well as the heading at which the vehicle starts to take a turn.

Several cases where the paths are potentially of the type $CCC$ are as shown in Fig. 4.7. Not that, for such cases, considering the initial and final turning segments, one has to check, if these turning segments intersect or not. If the segments intersect, then the potential candidate curve concatenation is of $CCC$

**Algorithm 2** *CCC Type Synthesis*

**Given** $q_i$, $q_f$, $\eta$, $\phi$, and $\rho$
**Simulate** $f(q, u, t)$ for $t \in [0, P]$ **with boundary condition** $f(q, \pm 1, 0) = q_i$ to get $C_i$
**Simulate** $f(q, u, t)$ for $t \in [0, P]$ **with boundary condition** $f(q, \pm 1, P) = q_f$ to get $C_f$

---

**for** Identical pairs of $C_i$ and $C_f$ **do**
  **Solve** for $t_{i,C_i,C_f}$, $t_{m,C_i,C_f}$, and $t_{f,C_i,C_f}$
  **Simulate** $f(q, u, t)$ **with boundary condition** $f(q, \pm 1, 0) = C_i(t)$ and $f(q, \pm 1, t < P) = C_f(t)$

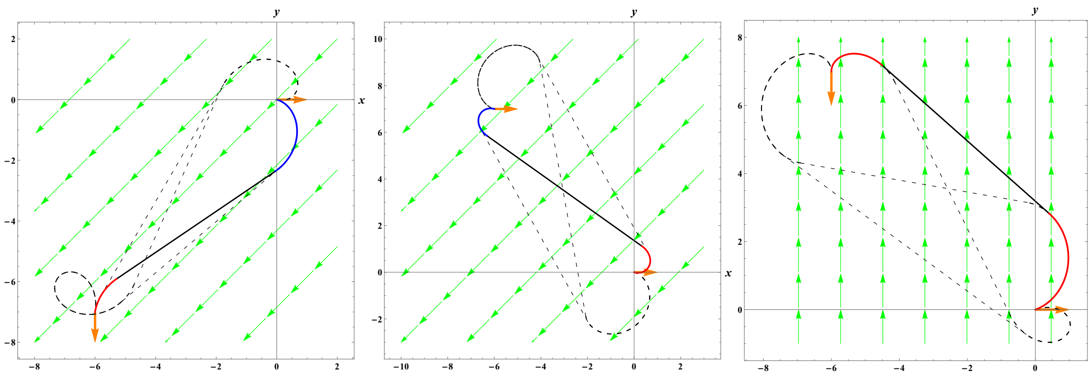
$$\begin{aligned} x_{C_m} &= x_{C_f} \\ y_{C_m} &= y_{C_f} \\ \tan^{-1}\left(\dot{x}_{C_m}, \dot{y}_{C_m}\right) &= \tan^{-1}\left(\dot{x}_{C_f}, \dot{y}_{C_f}\right) \end{aligned}$$

  **Compute** $t_{fC_iC_f} = P - t_{fC_iC_f}$
  **Compute** $T_{C_iC_f} = t_{iC_iC_f} + t_{mC_iC_f} + t_{fC_iC_f}$
**end for**
**FindMin** $T_{C_iC_f}$

type, whereas if the turning segment do not intersect, the potential candidate curve concatenation is of $CSC$ type. Referring to Fig. 4.7b, the candidate curve are $LRL$, $RLR$, as well as $LSL$, $RSR$, and $LSR$. In this particular case, the shortest path is the $RSR$. The proposed algorithm attempts to solve for all possible paths, and identified the time optimal one whether it is a $CCC$ or a $CSC$ type. As for Fig. 4.7a and Fig.4.7c, the time optimal path for both is of type $LRL$.

Fig. 4.8 compares the optimal paths for both the rotational and irrotational models where $\rho$ is assumed to equal 0.5. The dashed paths represent the optimal path corresponding to the *irrotational model* whereas the solid paths correspond to the *rotational model*. Accounting for a rotational effect in the model shows that it is possible to traverse turning curves with a smaller turning radius compared to the *irrotational model* as shown in the final left turn of Fig. 4.8a. On the other hand, for different scenarios such as the initial left turn of Fig. 4.8a or the final right turn of the path in Fig. 4.8b it is only possible to follow the corresponding curves with radii greater than those of the turns corresponding to the *irrotational model*.

In Fig. 4.8c the maneuver of parking in the same position but with an opposite direction is considered, that is, going from $q_i = (0, 0, 0)$ to $q_f = (0, 0, \pi)$. For this case, the optimal path is $LRL$ for both models. However, note that for the *irrotational model*, the solution path is symmetric about the $x$-axis. This, does not represent accurately the effect of a flow on a vehicle. As for the *rotational model*, the effect of the flow breaks the symmetry of the curve and yields a more reasonable solution path.

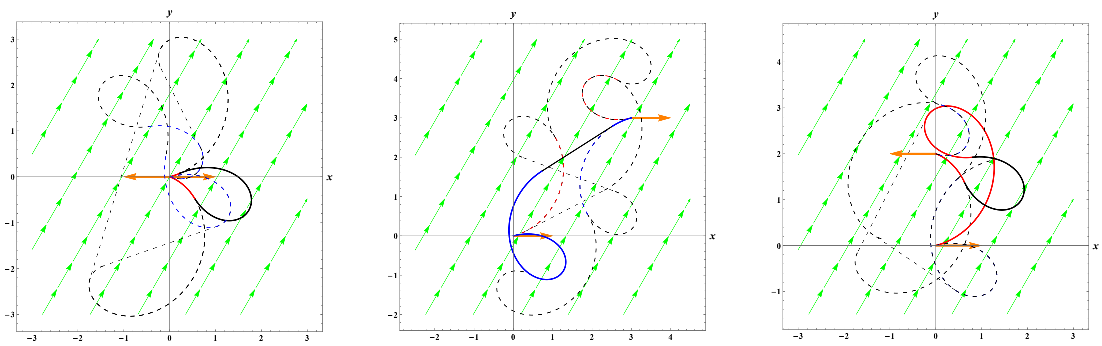


(a) Time Optimal Path for $\eta = 0.3$, $\phi = \frac{-3\pi}{4}$, $q_f = \{-6, -7, \frac{-\pi}{2}\}$ is $RSL$ with $t = 8.48s$

(b) Time Optimal Path for $\eta = 0.3$, $\phi = \frac{-3\pi}{4}$, $q_f = \{-6, 7, 0\}$ is $LSR$ with $t = 12.56s$

(c) Time Optimal Path for $\eta = 0.35$, $\phi = \frac{\pi}{2}$, $q_f = \{-6, 7, \frac{-\pi}{2}\}$ is $LSL$ with $t = 10.23s$

Figure 4.6: Time optimal paths for cases where only paths of type $CSC$ are possible

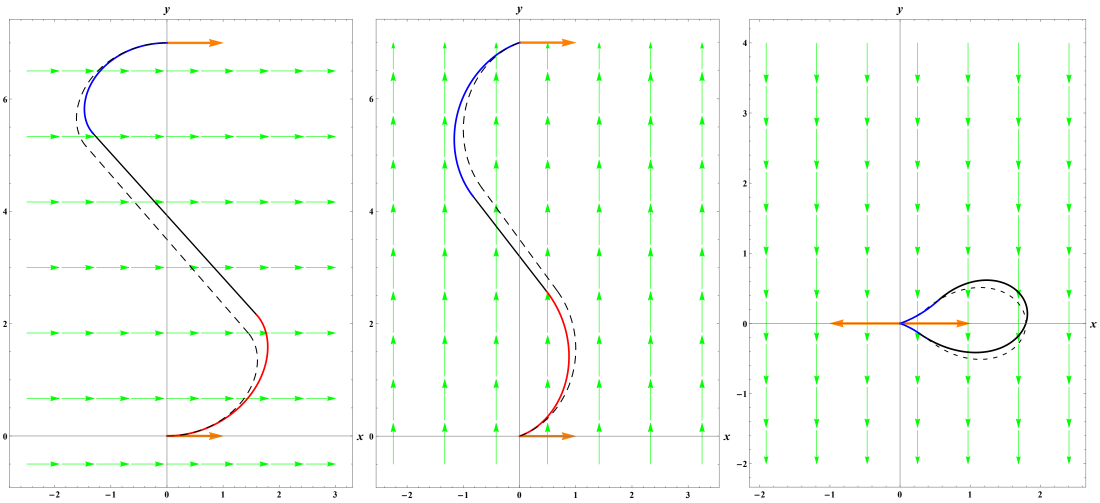


(a) Time Optimal Path for $q_f = \{0, 0, \pi\}$ is $LRL$ with $t = 4.98s$

(b) Time Optimal Path for $q_f = \{3, 3, 0\}$ is $RSR$ with $t = 7.93s$

(c) Time Optimal Path for $q_f = \{0, 2, \pi\}$ is $LRL$ with $t = 10.8s$

Figure 4.7: Cases where both types $CSC$ and $CCC$ are time optimal candidates for $\eta = 0.35$, $\phi = \frac{\pi}{3}$



(a) Time Optimal Path for $\eta = 0.35$, $\phi = 0$, $q_f = \{0, 7, 0\}$ is $LSR$ with $t = 11.12s$

(b) Time Optimal Path for $\eta = 0.35$, $\phi = \frac{\pi}{2}$, $q_f = \{0, 7, 0\}$ is $LSR$ with $t = 6.68s$

(c) Time Optimal Path for $\eta = 0.35$, $\phi = \frac{-\pi}{2}$, $q_f = \{0, 0, \pi\}$ is $LRL$ with $t = 5.83s$

Figure 4.8: Optimal paths for the *Rotational model* versus the *Irrotational model*

# Chapter 5

# Minimum Energy Trajectories and Future Work

The model developed and preliminary results obtained from solving the minimum time problem give a rise to many research questions. In a first attempt of pursuing paths that make the AUV go from initial to final configurations while using the minimum amount of energy or control input possible the problem is solved as a fixed end-time two point boundary value problem. In order to incorporate the final time required by the solver, a scaling of time is introduced and the final time is added as an augmented state to the system allowing the problem to be solved numerically. After that the minimum time and minimum energy results are compared when obtained using the same method. Then, observations regarding the bounds on the costates and the space of feasible solutions are presented. Finally, possible future directions to solve the problem are discussed. These directions include converting the problem to a nonlinear programming problem or writing the system in terms of body frame coordinates, solve the problem as an optimal control problem on *Lie Groups* then transform the solution back to the world or reference frame.

## 5.1 Optimal Control Formulation

For the system of main interest in this work, an Autonomous Underwater Vehicle, most of the energy expenditure is due to the forward propelling control input. Hence, the model is modified to include a variable control input $v(t)$ on the forward velocity (propelling) in addition to the previously introduced control on the steering, $u(t)$. The model then becomes,

$$
\begin{aligned}
\dot{x}(t) &= \nu(t)\cos\theta(t) + \eta\cos\phi, \\
\dot{y}(t) &= \nu(t)\sin\theta(t) + \eta\sin\phi, \\
\dot{\theta}(t) &= u(t) + \rho\eta\sin\big(\phi - \theta(t)\big)\,.
\end{aligned} \tag{5.1}
$$

A new cost function is introduced to incorporate both costs corresponding to the total time of travel and the the total expenditure of control inputs along the optimal trajectory. The new cost function penalizes the running cost of time and control inputs where $\alpha$ is the weight associated with the time of travel, $\beta$ is the weight associated with the forward velocity control input $v(t)$ and $\gamma$ is the weight of the steering control input $u(t)$, the general form of the cost function is then given by,

$$
J = \int_0^{t_f} \left(\alpha + \beta\nu^2 + \gamma u^2\right) dt \tag{5.2}
$$

The three positive constants $\alpha$, $\beta$, and $\gamma$ are positive constant weights selected by the user according to the system of interest, motor power consumption values, and total time of travel importance relative to that of control expenditure with respect to a certain mission. The control hamiltonian describing the system then becomes,

$$
\begin{aligned}
H =&\alpha + \beta\nu(t)^2 + \lambda_x(t)(\eta\cos(\phi) + \nu(t)\cos(\theta(t))) \\
&+ \lambda_y(t)(\eta\sin(\phi) + \nu(t)\sin(\theta(t))) + \gamma u(t)^2 \\
&+ \lambda_\theta(t)(\eta\rho\sin(\phi - \theta(t)) + u(t))
\end{aligned} \tag{5.3}
$$

Where both control functions $u(t)$ and $v(t)$ appear as quadratic terms in the hamiltonian which makes it possible to solve for the optimal controls that will minimize the hamiltonian with respect to the control variables $u(t)$ and $v(t)$. Consider the partial derivatives of the hamiltonian with respect to control inputs,

$$
0 = \frac{\partial H}{\partial u} = \lambda_\theta(t) + 2\gamma u(t) \tag{5.4}
$$

$$
0 = \frac{\partial H}{\partial v} = 2\beta\nu(t) + \lambda_x(t)\cos(\theta(t)) + \lambda_y(t)\sin(\theta(t)) \tag{5.5}
$$

resulting in the equations describing the controls that minimize the hamiltonian

$$
u^*(t) = -\frac{\lambda_\theta(t)}{2\gamma} \tag{5.6}
$$

$$
v^*(t) = -\frac{1}{2\beta}\left(\lambda_x(t)\cos(\theta(t)) + \lambda_y(t)\sin(\theta(t))\right) \tag{5.7}
$$

Also it is important to note that the second derivatives of the hamiltonian indicate that the resulting optimal control solutions actually minimize the hamiltonian and do not describe a maximum or a saddle point. The second derivatives turn out to be positive in value indicating a minimum in the hamiltonian.

$$\frac{\partial^2 H}{\partial u^2} = 2\gamma \tag{5.8}$$

$$\frac{\partial^2 H}{\partial v^2} = 2\beta \tag{5.9}$$

Moreover, recall that the equations describing the dynamics of the costates of the system are given by

$$\begin{aligned}
\dot{\lambda}_x &= -\frac{\partial H}{\partial x} =0 \\
\dot{\lambda}_y &= -\frac{\partial H}{\partial y} =0 \\
\dot{\lambda}_\theta &= -\frac{\partial H}{\partial \theta} =\eta\rho\lambda_\theta(t)\cos(\phi-\theta(t)) + \lambda_x(t)v(t)\sin(\theta(t)) \\
&\qquad - \lambda_y(t)v(t)\cos(\theta(t))
\end{aligned} \tag{5.10}$$

Substituting the equations for optimal controls obtained in (5.6) and (5.7) into the state and costate dynamics described in (5.1) and (5.10) creates a system of six coupled nonlinear ordinary differential equations for the dependent variables $\widetilde{q} = \big\{x(t), y(t), \theta(t), \lambda_x(t), \lambda_y(t), \lambda_\theta(t)\big\}$. The resulting system is then given by,

$$\begin{aligned}
\dot{x}(t) &= \frac{-\cos\big(\theta(t)\big)\Big(\lambda_x(t)\cos\big(\theta(t)\big) + \lambda_y(t)\sin\big(\theta(t)\big)\Big)}{2\beta} + \cos(\phi) \\
\dot{y}(t) &= \frac{-\sin\big(\theta(t)\big)\Big(\lambda_x(t)\cos\big(\theta(t)\big) + \lambda_y(t)\sin\big(\theta(t)\big)\Big)}{2\beta} + \sin(\phi) \\
\dot{\theta}(t) &= \eta\rho\sin\big(\phi-\theta(t)\big) - \frac{\lambda_\theta(t)}{2\gamma} \\
\dot{\lambda}_x(t) &= 0 \\
\dot{\lambda}_y(t) &= 0 \\
\dot{\lambda}_\theta(t) &= \frac{1}{4\beta}\Big(4\beta\eta\rho\lambda_\theta(t)\cos(\phi-\theta(t)) + \sin(2\theta(t))\big(\lambda_y(t)^2 - \lambda_x(t)^2\big) \\
&\qquad + 2\lambda_x(t)\lambda_y(t)\cos(2\theta(t))\Big)
\end{aligned} \tag{5.11}$$

Knowing the initial conditions on all six dependent variables of $\widetilde{q}$ and the total time of travel $t_f$ makes it possible to simply integrate the set of first order differential equations. This will generate the trajectory along with the controls required to traverse it. However, for the proposed path planning problem, the boundary conditions are only imposed on the state variables, namely $q = \{x(t), y(t), \theta(t)\}$. Moreover, the problem is a free end-time problem meaning that $t_f$ is not specified.

Thus, solving the minimum energy path planning problem requires solving the *Two Point Non-Linear Boundary Value Problem* ($BVP$) where the independent variable $t$ is also unknown at the final boundary, meaning that it is also required to solve for $t_f$ that will minimize the provided cost functional $J$ provided in (5.2).

A *Two Point BVP* where $t_f$ is specified is known as a *Standard Two Point BVP* and few numerical algorithms are available to solve such problems. The most common algorithm used is known as the *Shooting Algorithm* where an initial guess is taken for the remaining initial conditions required to integrate the set of differential equations and the system is then integrated over the interval of the independent variable $t \in [0, t_f]$. The values for the dependent variables at $t_f$ are recorded and then the initial guess is modified accordingly until the boundary condition at $t_f$ has been satisfied.

## 5.2 Manually Assigning $t_f$

As a first Attempt to gain insight and solve the minimum energy path planning problem, consider the case where there is no time cost and the control inputs are unbounded. In other words, the time of travel is not penalized in the cost function, hence $\alpha = 0$ and the control inputs $u(t)$ and $v(t)$ can take any value. Keeping in mind that the motor propelling the vehicle will consume more power than the actuators used for steering, let the weights associated with the control inputs in the cost function be $\beta = 1$ and $\gamma = 0.5$. Then, the *Standard Two Point BVP* is solved for fixed boundary conditions, i.e. fixed initial and final configurations $q_i$ and $q_f$, fixed flow conditions $\rho$, $\eta$ and $\phi$ and fixed $t_f$.

Through out the examples presented below, consider the case where $q_i = 0, 0, 0$ and $q_f = 7, 5, 0$. The flow conditions are given by $\rho = 0.2$, $\eta = 0.3$ and $\phi = \frac{\pi}{4}$. The problem is solved for different values of $t_f$ and few samples of the obtained trajectories are presented below along with the control inputs used to generate them.

### 5.2.1 Case 1: $t_f = 1$

For the first case, time is too short and the control distance cannot be covered by the systems for control inputs that are bounded. it is clear that the control input required to cover the distance in a very short time will exceed the control bounds. This leads to a very high value of the cost function which defies the purpose of trying to minimize the control expenditure, hence a longer time of travel is required for the system to go from an initial to final configuration.

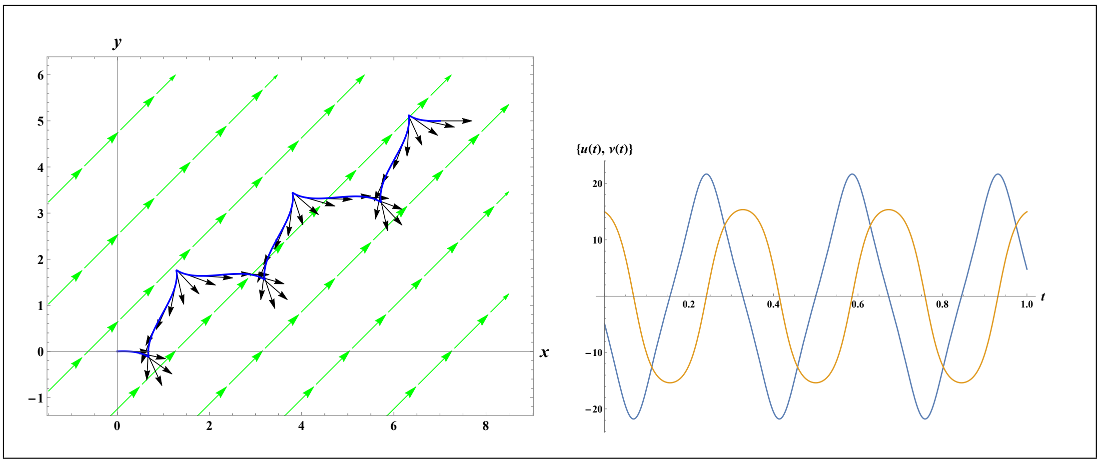



Figure 5.1: **case 1** $t_f = 1$ and $J = 236.117$

### 5.2.2 Case 2: $t_f = 40$

For the second case a total travel time is selected to be $t_f = 40$. By inspecting the results, it is clear that the control input stays within bounds. However, the cost function retains a relatively high value at $J = 64.8377$. Also, it appears that the solution for this case includes periodic steering in order to traverse a longer trajectory to satisfy the total time of travel constraint. Meaning that the system traverses a longer trajectory in order to spend more time before arriving at the final configuration, but by doing this excess energy is being spent. Hence a shorter total time of travel is required to reach the final state.

### 5.2.3 Case 3: $t_f = 7.5$

The third case where $t_f = 7.5$ yields a more reasonable result with a more satisfying total cost $J = 0.432$ and with controls that remain withing the bounded input intervals. The trajectory traversed by the system matches the intuition where the vehicle or robot drifts with the flow fields while applying the minimum

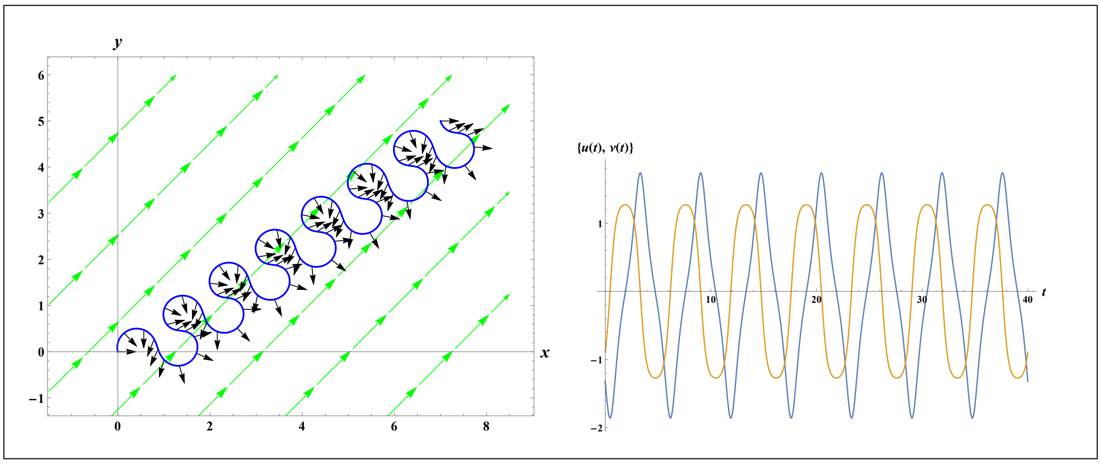



Figure 5.2: **case 2** $t_f = 40$ and $J = 64.8377$

possible control input to end up at the desired final configuration $q_f$.

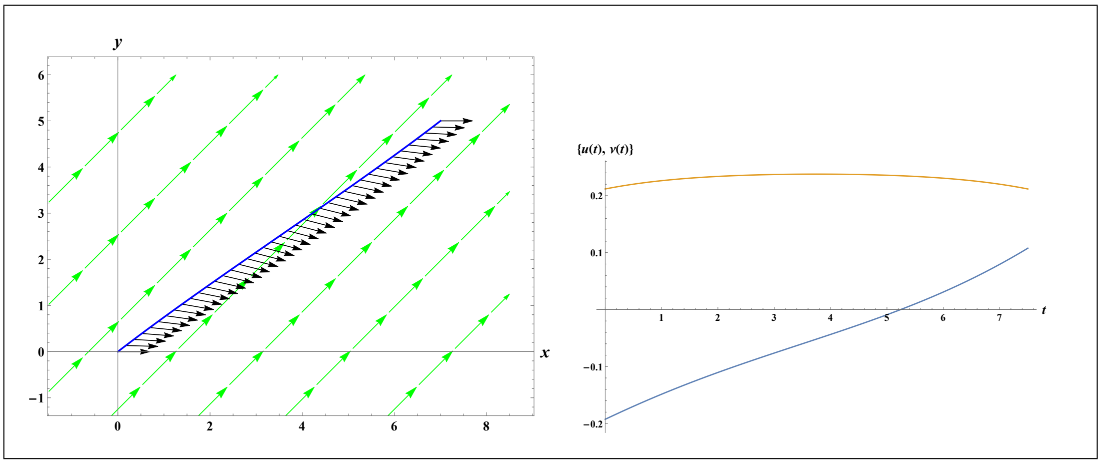



Figure 5.3: **case 3** $t_f = 7.5$ and $J = 0.432$

More cases were simulated and the resulting total cost and trajectory lengths are presented in the the below two figures 5.4 and 5.5 respectively. It is obvious that the choice of final time affects the total cost of travel. However, it is not clear if a certain final time yields a global minimum or if a global minimum actually exists. Also, the total travel time affects the total length of the trajectory obtained by the solver but no clear relation can be concluded between the total length $L$ and the total time of travel $t_f$. By looking at the results obtained one can conclude that more than one solution may satisfy the conditions imposed by *Pontryagin's minimum principle*. However, this doesn't guarantee the convergence of the solver to an actual global optimal solution.

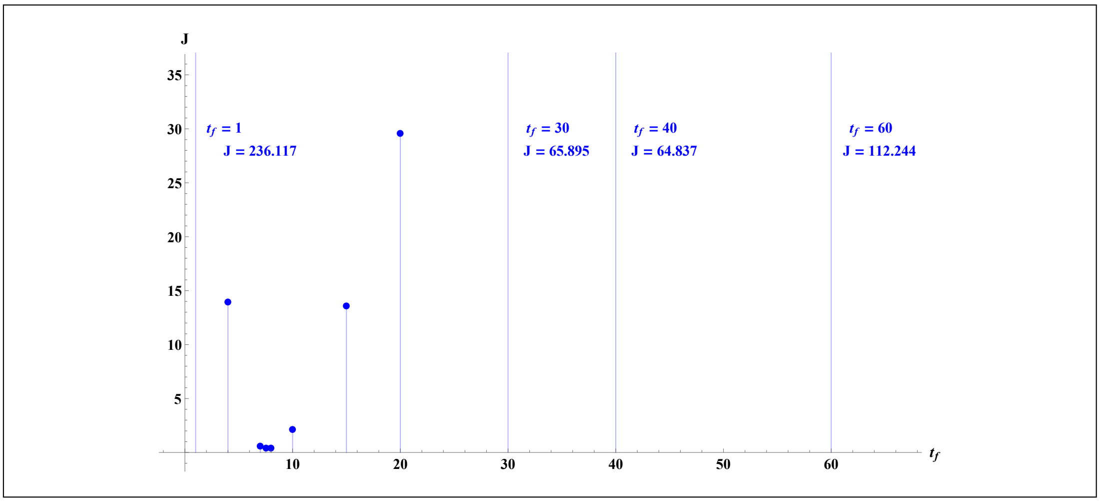



Figure 5.4: Total travel cost $J$ for various total travel time $t_f$

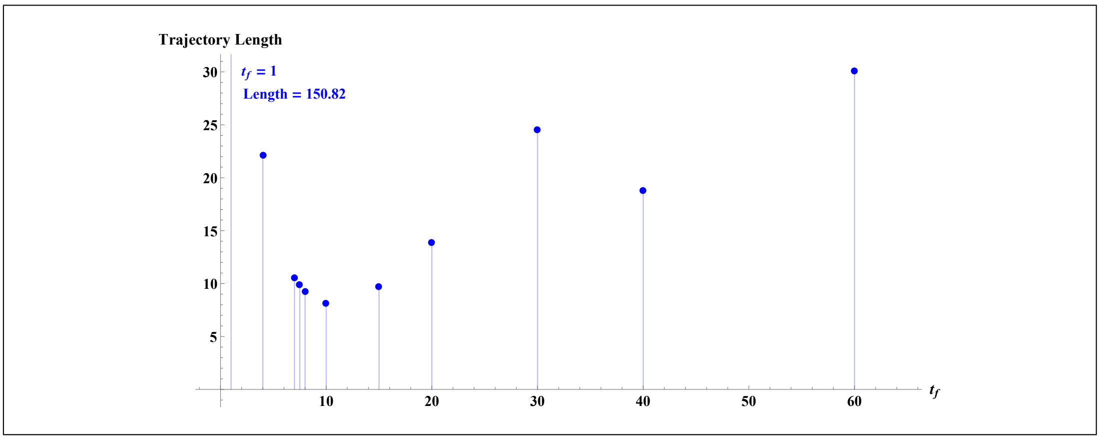



Figure 5.5: Total trajectory length $L$ for various total travel time $t_f$

## 5.3 A Standard Form Two Point Boundary Value Problem

It is clear that the total time of travel affects the cost of travel and the shape of the trajectory traversed. It also affects the value of the control input required to generate such trajectories. Selecting $t_f$ manually comes out to be an ineffective approach for solving the problem and doesn't guarantee an optimal solution. An approach proposed in [48] takes care of the problem of finding a suitible final time.

First a scaling of time is introduced by

$$\tau = \frac{t}{t_f} \tag{5.12}$$

$$d\tau = \frac{dt}{t_f} \tag{5.13}$$

The second step is to add a dummy state $r(t)$ that corresponds to $t_f$ with the trivial dynamics $\dot{r}(t) = 0$. Then the new system is derived from the system defined in (5.1) and (5.10) by

$$\dot{q}(t) = f\left(q, u, v, t\right) \Rightarrow \dot{y}(t) = t_f f\left(q, u, v, \tau\right) \tag{5.14}$$

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial q} \Rightarrow \dot{\Lambda}(t) = -t_f \frac{\partial H}{\partial q} \tag{5.15}$$

Adding a fourth state $r(t)$ the the newly derived system results in a system of seven coupled non-linear differential equations. Note that it is inefficient to add a fourth costate for the state $r(t)$ since it won't be appearing anywhere in the system. The final time is now set to, or determined to be $t_f = 1$. Then the new system can be solved as a *Standard Form Two Point BVP*. The resulting standard form system is given by,

$$\begin{aligned}
\dot{x}(t) &= r(t)\Bigg(\frac{-\cos\big(\theta(t)\big)\Big(\lambda_x(t)\cos\big(\theta(t)\big) + \lambda_y(t)\sin\big(\theta(t)\big)\Big)}{2\beta} + \cos(\phi)\Bigg) \\
\dot{y}(t) &= r(t)\Bigg(\frac{-\sin\big(\theta(t)\big)\Big(\lambda_x(t)\cos\big(\theta(t)\big) + \lambda_y(t)\sin\big(\theta(t)\big)\Big)}{2\beta} + \sin(\phi)\Bigg) \\
\dot{\theta}(t) &= r(t)\Big(\eta\rho\sin\big(\phi - \theta(t)\big) - \frac{\lambda_\theta(t)}{2\gamma}\Big) \\
\dot{r}(t) &= 0 \\
\dot{\lambda}_x(t) &= 0 \\
\dot{\lambda}_y(t) &= 0 \\
\dot{\lambda}_\theta(t) &= \frac{r(t)}{4\beta}\Big(4\beta\eta\rho\lambda_\theta(t)\cos(\phi - \theta(t)) + \sin(2\theta(t))\big(\lambda_y(t)^2 - \lambda_x(t)^2\big) \\
&\quad + 2\lambda_x(t)\lambda_y(t)\cos(2\theta(t))\Big)
\end{aligned} \tag{5.16}$$

In order to solve this standard problem one more boundary condition is needed. This boundary condition is provided by the boundary condition on the hamiltonian at the final time $t_f = 1$ for the re-scaled problem and given by,

$$
\begin{aligned}
0 = H = &\alpha + \beta\nu(1)^2 + r(1)\lambda_x(1)(\eta\cos(\phi) + \nu(1)\cos(\theta(1))) \\
&+ r(1)\lambda_y(1)(\eta\sin(\phi) + \nu(1)\sin(\theta(1))) + \gamma u(1)^2 \\
&+ r(1)\lambda_\theta(1)(\eta\rho\sin(\phi - \theta(1)) + u(1))
\end{aligned} \tag{5.17}
$$

This method of converting the *free end time problem* to a *standard form problem* is implemented and the results obtained are presented below.

### 5.3.1 Case 1: Unbounded Control Input

The standard form system derived in eqn. (5.16) is solved for the same boundary conditions presented at the beginning of this chapter along with the boundary condition from the hamiltonian at the final time in eqn. (5.17). First both control inputs are assumed to be unbounded and can take any value. The solver is able to converge to a solution satisfying the state and costate dynamics along with the optimal controls solution obtained by *PMP*. The obtained solution provides a low total travel cost at $J = 1.76$, a relatively short total time of travel $t_f = 2.566$. However the resulting control inputs exceed the bounds or the constraints on the total control inputs allowed. to solve this problem, the control constraints are imposed on the system and the solution is obtained using the same method in the next section.

### 5.3.2 Case 2: Bounded Control Input

In the previous section the solver converged to a solution satisfying all the constraints and dynamics equations except for the bounds imposed on the control inputs. To enforce a solution where the control inputs are bounded and given by,

$$u(t) \in [-1, 1] \tag{5.18}$$

$$\nu(t) \in [0, 1] \tag{5.19}$$

The unbounded control functions are replaced by new bounded control inputs $U(t)$ and $V(t)$ where,

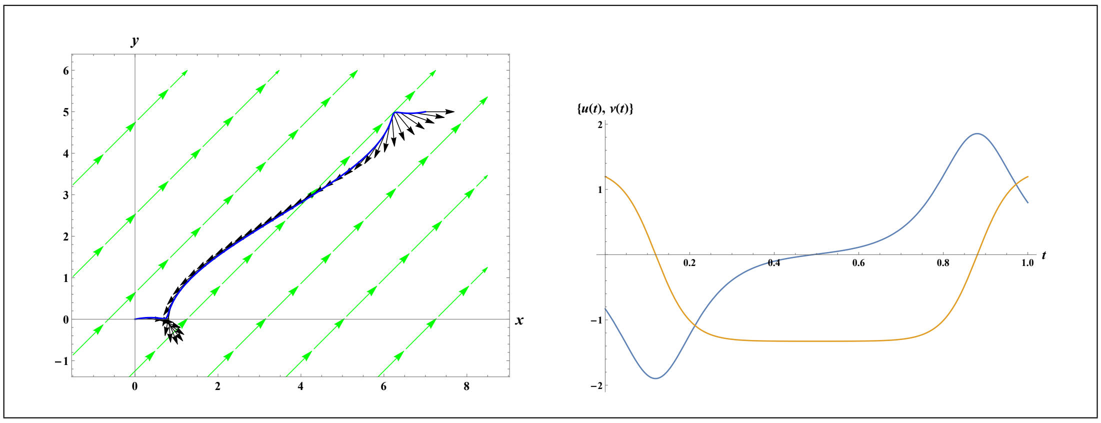



Figure 5.6: **case 1** Normalized final time $t_f$ with unbounded control inputs, $J = 1.76$ and $t_f = 2.566$

$$U^*(t) = \begin{cases} -1 & \text{if } u^*(t) < -1 \\ 1 & \text{if } u^*(t) > 1 \\ u^*(t) & \text{otherwise} \end{cases} \tag{5.20}$$

$$V^*(t) = \begin{cases} 0 & \text{if } \nu^*(t) < 0 \\ 1 & \text{if } \nu^*(t) > 1 \\ \nu^*(t) & \text{otherwise} \end{cases} \tag{5.21}$$

The results come out to be more satisfying than the case where no bounds where imposed on the control inputs of the system. The total travel cost attains a very low value at $J = 0.00316$ and a reasonably moderate total time of travel $t_f = 6.565$. The behavior of the system matches the intuition of drifting with the flow with using the minimum control possible to reach the final desired configuration. However, obtaining two different solutions for the system satisfying the optimal control and PMP's formlation raises many questions regarding the global optimality of the obtained solutions.

## 5.4 Analysis of Bounded Control Inputs

The shooting algorithm was used to obtain numerical solutions for all the results obtained in sections 5.2, 5.3, and 5.5. The shooting algorithm however is time consuming and fails to converge to a solution in some cases due to the high nonlinearity of the problem. To speed up the time it takes for the shooting algorithm to converge to a solution some bounds are imposed on the dependent

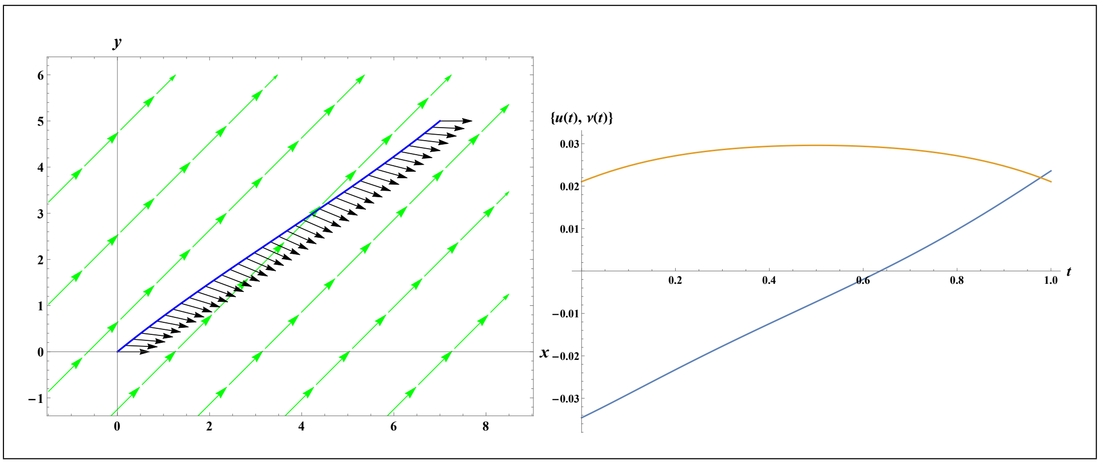



Figure 5.7: **case 2** Normalized final time $t_f$ with bounded control inputs, $J = 0.000896$ and $t_f = 7.7233$

variables of the two point boundary value problem.

### 5.4.1 Bounds on $\lambda_x$ and $\lambda_y$

In order to obtain bounds on the costates $\lambda_x(t)$ and $\lambda_y(t)$ the optimal control $v^*(t)$ is studied. The equation describing $v^*(t)$ in terms of the states and costates is given in eqn. (5.7) by

$$v^*(t) = -\frac{1}{2\beta}\left(\lambda_x(t)\cos(\theta(t)) + \lambda_y(t)\sin(\theta(t))\right) \tag{5.22}$$

$\beta$ is a nonzero positive constant that is specified by the user. Also, $v^*(t) \in [0, 1]$. Interesting cases rise in equation (5.22), at $\theta(t) = 0, \frac{\pi}{2}, \pi, \frac{-\pi}{2}$

**Case where $\theta(t) = 0$:**

For $\theta(t) = 0$ the equation describing the optimal propelling control becomes

$$v^*(t) = -\frac{\lambda_x(t)}{2\beta}, \tag{5.23}$$

if $v^*(t) = 0$ then $\lambda_x(t) = 0$. Else if $v^*(t) = 1$ then $\lambda_x(t) = -2\beta$.

**Case where $\theta(t) = \frac{\pi}{2}$:**

For $\theta(t) = \frac{\pi}{2}$ the equation describing the optimal propelling control becomes

$$v^*(t) = -\frac{\lambda_y(t)}{2\beta}, \tag{5.24}$$

if $v^*(t) = 0$ then $\lambda_y(t) = 0$. Else if $v^*(t) = 1$ then $\lambda_y(t) = -2\beta$.

**Case where $\theta(t) = \pi$:**

For $\theta(t) = \pi$ the equation describing the optimal propelling control becomes

$$v^*(t) = \frac{\lambda_x(t)}{2\beta}, \tag{5.25}$$

if $v^*(t) = 0$ then $\lambda_x(t) = 0$. Else if $v^*(t) = 1$ then $\lambda_y(t) = 2\beta$.

**Case where $\theta(t) = \frac{-\pi}{2}$:**

For $\theta(t) = -\frac{\pi}{2}$ the equation describing the optimal propelling control becomes

$$v^*(t) = \frac{\lambda_y(t)}{2\beta}, \tag{5.26}$$

if $v^*(t) = 0$ then $\lambda_y(t) = 0$. Else if $v^*(t) = 1$ then $\lambda_y(t) = 2\beta$. Hence, from the cases studied above,

$$\lambda_x(t) \text{ } and \text{ } \lambda_y(t) \in [-2\beta, 2\beta]\,. \tag{5.27}$$

### 5.4.2 Bounds on $\lambda_\theta$

The equation describing the optimal steering control input is given by eqn. (5.6) as

$$u^*(t) = -\frac{\lambda_\theta(t)}{2\gamma}. \tag{5.28}$$

For the case where $u^*(t) = 1$ then $\lambda_\theta^*(t) = -2\gamma$. Else, when $u^*(t) = -1$ then $\lambda_\theta^*(t) = 2\gamma$. Now that the values the costates can attain are bounded by the bounds on the control inputs, it is possible to study the geometric structure relating these costates to the states.

$$\lambda_\theta(t) \in [-2\gamma, 2\gamma] \,. \tag{5.29}$$

### 5.4.3 Geometric Interpretation

Going back to the optimal control hamiltonian $H^*$. A boundary condition provided by PMP is that $H^*(t) = 0$. This condition makes it possible to solve for $\lambda_\theta^*(t)$

$$\lambda_\theta^*(t) = 2\left(\gamma\ \eta\ \rho\ \sin\left(\phi - \theta(t)\right) \pm \gamma\sqrt{a+b}\right) \tag{5.30}$$

where

$$a = \eta^2\rho^2\sin^2(\phi - \theta(t)) \tag{5.31}$$

and

$$\begin{aligned} b =& \frac{1}{4\beta\gamma}(-4\alpha\beta + \lambda_x(t)(\lambda_y(t)\sin(2\theta(t)) - 4\beta\eta\cos(\phi)) \\ &- 4\beta\eta\lambda_y(t)\sin(\phi) + \lambda_x(t)^2\cos^2(\theta(t)) + \lambda_y(t)^2\sin^2(\theta(t))) \end{aligned} \tag{5.32}$$

Now that $\lambda_\theta^*(t)$ is written in terms of the two constant costates $\lambda_x^*(t)$ and $\lambda_y^*(t)$ then it is possible to substitute it back as the optimal steering control $u^*(t)$. The optimal steering control eliminates the drift and leaves an equation describing turning rate dynamics as follows,

$$\dot\theta^*(t) = \pm\sqrt{a-b} \tag{5.33}$$

The fact that the optimal steering rate contains a square root imposes more constraints on the costates and states relation. This means that

$$a - b \geqslant 0 \tag{5.34}$$

To visualize the relation between the costates $\lambda_x^*(t)$, $\lambda_y^*(t)$ and $\theta^*(t)$ depending on the flow conditions $\rho$, $\eta$, and $\phi$ it is possible to generate contour plots of the regions satisfying the condition in eqn. (5.34) within the bounds obtained earlier on $\lambda_x^*(t)$ and $\lambda_y^*(t)$ as shown in figure 5.8. Assuming similar flow condition to the ones considered earlier in this chapter, and with $\alpha = 0$, $\beta = 1$, and $\gamma = 0.5$ provides an example of a resulting plot of the region satisfying the condition provided in eqn. (5.34). This geometric visualization shows the region where a value for $\dot{\theta}^*(t)$ always exists.

For a motion planning problem, two planes defining the two constant costates $\lambda_x^*(t)$ and $\lambda_y^*(t)$ could be constructed. The line resulting from the intersection of the two constant planes must connect both the initial and final heading angles $\theta(t_0)$ and $\theta(t_f)$ and remain entirely within the region satisfying the existence condition provided earlier in eqn. (5.34).

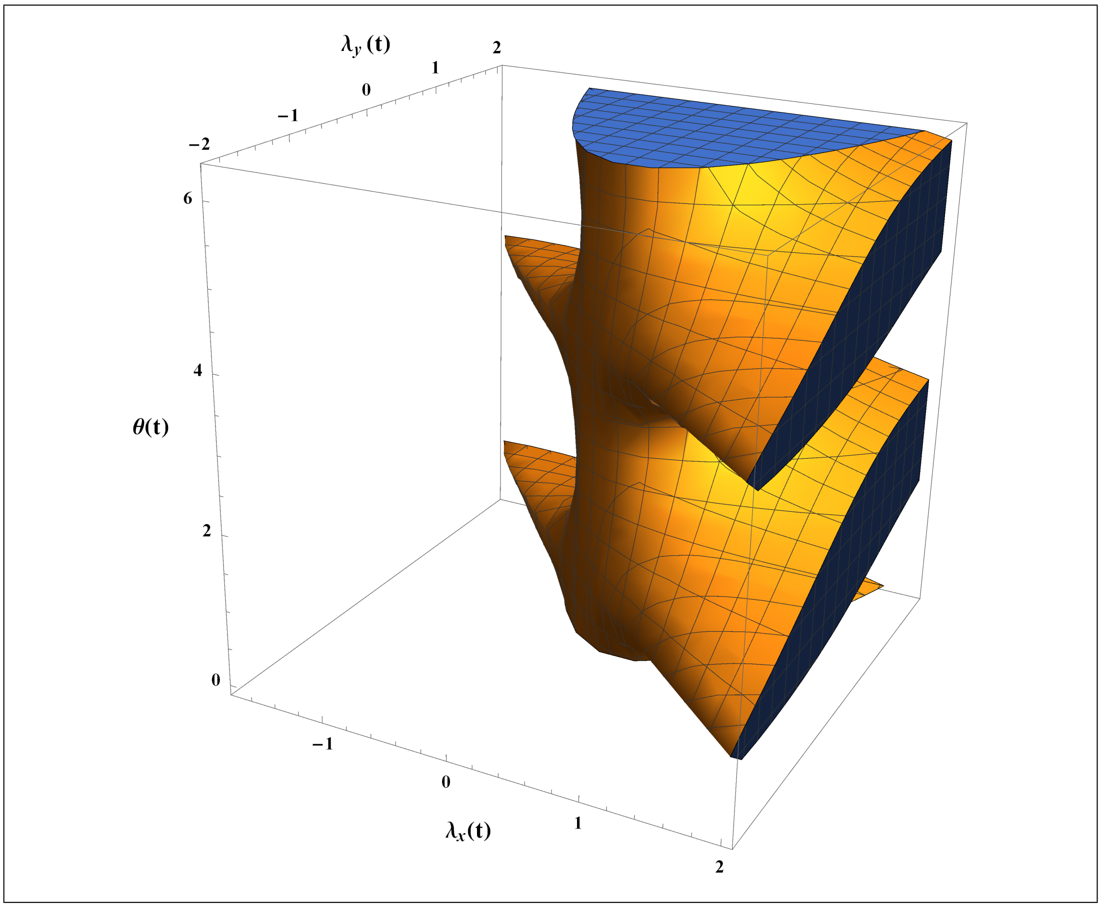



Figure 5.8: A region plot of the values satisfying the existence condition of $\dot{\theta}^*(t)$

### 5.4.4 Results

In the previous sections of this chapter both the flow parameters and final configuration were fixed and various methods to solve the problem were explored. In section 5.3 a method that guarantees the bounds on the controls and solves for the final time of the problem was found. This method solves the two point boundary value problem resulting form Pontryagin's minimum principle.

Some results of the method derived above where the controls are bounded and the final time is optimized are presented in this section. The initial configuration is assumed to be $q_i = \{0, 0, 0\}$ for all cases. Also, both $\eta$ and $\rho$ are assumed to equal 0.3. Figure 5.9 depicts the case where the flow is directed at $\phi = \frac{\pi}{2}$ and the target configuration is $q_f = \{5, 3, 0\}$. For this case the cost comes out to be $J = 1.88$ while the total duration spent traversing the trajectory is $t_f = 17.547$.

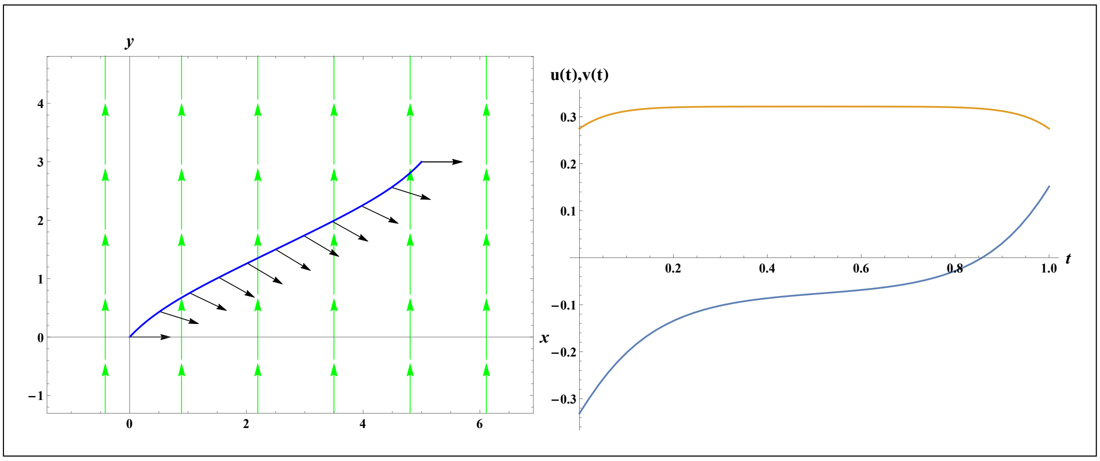



Figure 5.9: $\phi = \frac{\pi}{2}$, $q_f = \{5, 3, 0\}$, $J = 1.88$ and $t_f = 17.547$

Figure 5.10 shows the case where the flow is directed at $\phi = -\frac{\pi}{2}$ and the final configurations is at $q_f = \{5, -5, 0\}$. The total control expenditure cost comes out to be $J = 1.376$ and the total time of travel along the trajectory is $t_f = 21.895$.

Figure 5.11 presents a case where the flow is also directed at $\phi = -\frac{\pi}{2}$, however the final configuration is given by $q_f = \{5, -5, -\frac{\pi}{2}\}$. The resulting structure of the optimal trajectory comes out to be completely different from the case presented in figure 5.10. The resulting trajectory has a total cost of $J = 1.79$ and a total traveling time of $t_f = 22.9$.

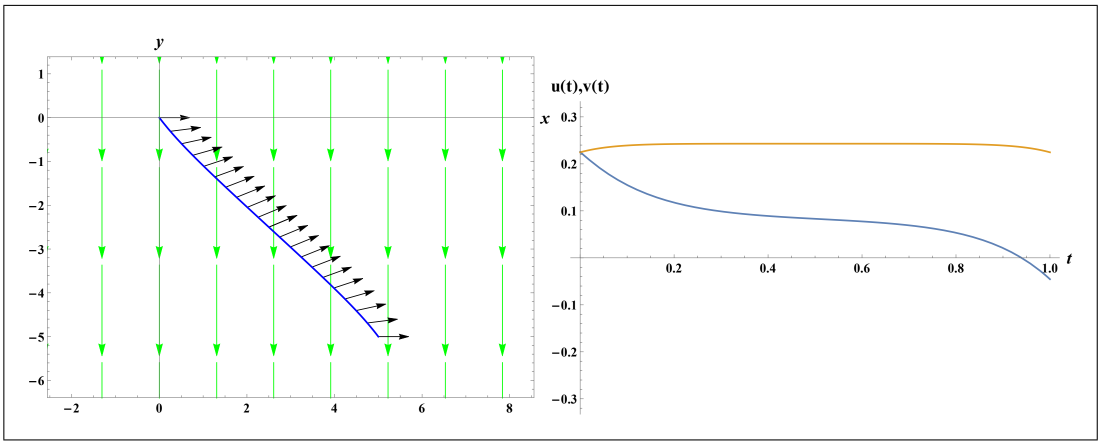



Figure 5.10: $\phi = -\frac{\pi}{2}$, $q_f = \{5, -5, 0\}$, $J = 1.376$ and $t_f = 21.895$

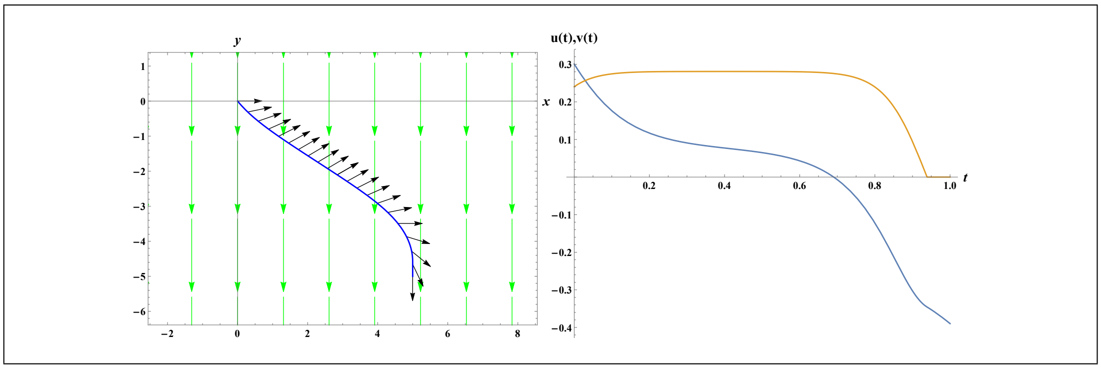



Figure 5.11: $\phi = -\frac{\pi}{2}$, $q_f = \{5, -5, -\frac{\pi}{2}\}$, $J = 1.79$ and $t_f = 22.9$

## 5.5 Penalizing Time

In order to compare the problem with the results of the minimum time problem, the term $\alpha$ associated with the cost of the time of travel is then increased while the weights associated with the control inputs are decreased. The results obtained show that as $\frac{\beta+\gamma}{\alpha} \to 0$ then the numerical solution converges to the minimum time solutions obtained in Chapter 4.

### 5.5.1 Case 1: Time and control input with same weights

The solution presented in this section uses a modified cost function. In order to penalize the total time of travel, the weight associated with the time running cost is set to $\alpha = 1$. The weights associated with the control inputs are kept the same with $\beta = 1$ and $\gamma = 0.5$. Figure 5.13 shows that the solver converges to a solution similar in structure to that of the case where the cost of the time of travel is not taken into consideration. However the control input cost $J = 1.005$ comes out to be higher in value and the total time of travel $t_f = 6.263$ is less than the case presented in section 5.3.2.

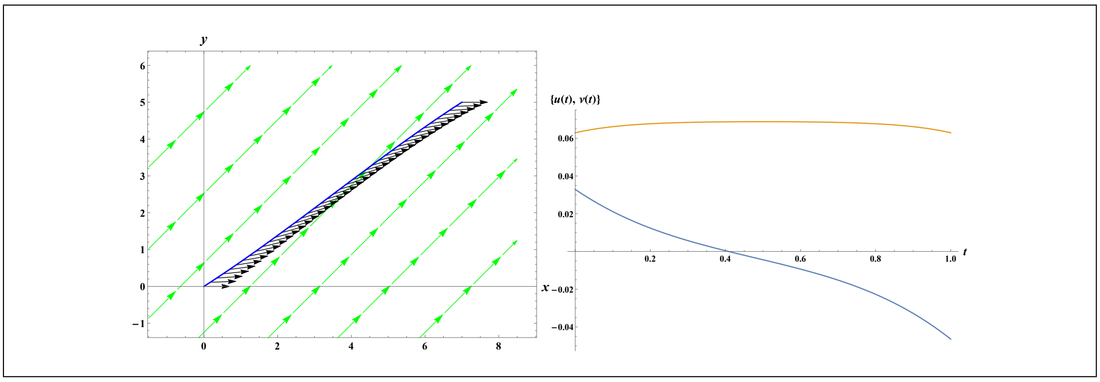



Figure 5.12: **case 1** where $\alpha = 1$, $\beta = 1$, $\gamma = 0.5$, $J = 1.005$, and $t_f = 6.263$

### 5.5.2 Case 2: Total time of travel with a very high cost

The weights of the control inputs cost is set to $\beta = \gamma = 0.0001$. This assigns a much higher cost to the total time of travel where $\alpha = 1$. Then the same approach used to solve the previous examples is used. The resulting trajectory comes out to be similar in structure to the results obtained when solving the minimum time problem in Chapter 4 as figure 5.13 indicates. The results turn out to be as expected, a decrease in the total time of travel $t_f = 4.395$. The control expenditure decreased in value and the total travel cost remained almost unchanged at $J = 1$.

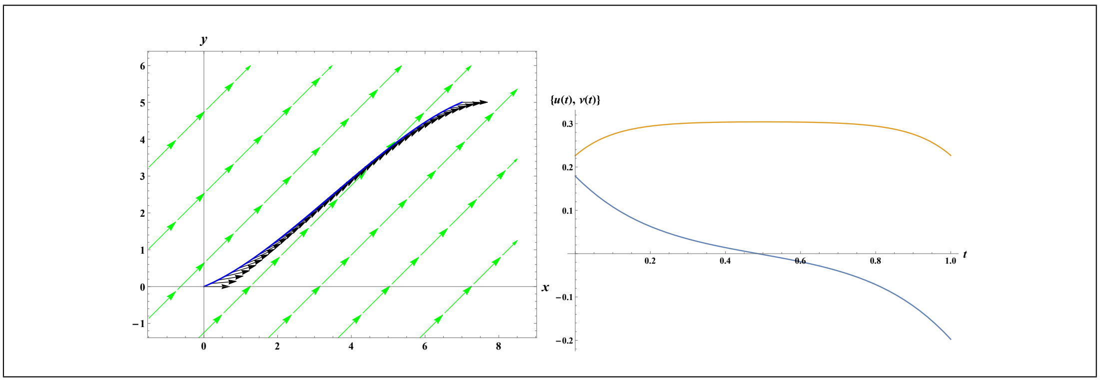



Figure 5.13: **case 2** where $\alpha = 1$, $\beta = \gamma = 0.0001$, $J = 1$, and $t_f = 4.395$

## 5.6 Future Work

This work started by introducing a new model to describe more accurately the motion of vehicles in certain mediums with flow fields. Next, a minimum time formulation was presented and a solution for the minimum time path planning was derived. Then a minimum energy path planning problem was formulated for the same model. Various methods to solve the minimum energy path planning problem were exploited and a geometric interpretation was presented. The work presented in this documents provided insight to optimal control problems including systems with drift. It shows how complicated it can be to arrive at an exact analytical or closed form solution for such problems. Never the less it raises many questions for future exploration. One of many future directions possible based on the results presented in this thesis is to develop the path synthesis algorithms presented in chapter 4 into much faster algorithms that could be implemented in real time. Another direction that could be taken is to develop numerical algorithms tailored to problems of this type that guarantee quick convergence to results satisfying the conditions provided by *Pontryagin's Minimum Principle*. One possible way to do this is presented by Bhattacharya [49] where the optimal control problem is converted to a discretized nonlinear programming problem. A completely different path could also be taken in solving such problems as inspired by Maclean [34] and Jurdjevic [50]. A work of this type will definitely be considered incomplete if not implemented in real life mobile robots and proved to deliver optimal performance.

# Bibliography


[1] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[2] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[3] S. M. LaValle, *Planning algorithms.* Cambridge university press, 2006.

[4] A. A. Markov, "Some examples of the solution of a special kind of problem on greatest and least quantities," *Soobshch. Karkovsk. Mat. Obshch*, vol. 1, pp. 250–276, 1887.

[5] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of mathematics*, pp. 497–516, 1957.

[6] J.-D. Boissonnat, A. Cérézo, and J. Leblond, *Shortest paths of bounded curvature in the plane.* Springer, 1993.

[7] X.-N. Bui, J.-D. Boissonnat, P. Soueres, and J.-P. Laumond, "Shortest path synthesis for dubins non-holonomic robot," in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 2–7, IEEE, 1994.

[8] J. Reeds and L. Shepp, "Optimal paths for a car that goes both forwards and backwards," *Pacific journal of mathematics*, vol. 145, no. 2, pp. 367–393, 1990.

[9] P. Soueres and J.-P. Laumond, "Shortest paths synthesis for a car-like robot," *IEEE Transactions on Automatic Control*, vol. 41, no. 5, pp. 672–688, 1996.

[10] D. J. Balkcom and M. T. Mason, "Time optimal trajectories for bounded velocity differential drive vehicles," *The International Journal of Robotics Research*, vol. 21, no. 3, pp. 199–217, 2002.

[11] D. J. Balkcom, P. A. Kavathekar, and M. T. Mason, "Time-optimal trajectories for an omni-directional vehicle," *The International Journal of Robotics Research*, vol. 25, no. 10, pp. 985–999, 2006.

[12] A. A. Furtuna and D. J. Balkcom, "Generalizing dubins curves: Minimum-time sequences of body-fixed rotations and translations in the plane," *The International Journal of Robotics Research*, 2010.

[13] H. Chitsaz and S. M. LaValle, "Time-optimal paths for a dubins airplane," in *Decision and Control, 2007 46th IEEE Conference on*, pp. 2379–2384, IEEE, 2007.

[14] T. G. McGee and J. K. Hedrick, "Optimal path planning with a kinematic airplane model," *Journal of guidance, control, and dynamics*, vol. 30, no. 2, pp. 629–633, 2007.

[15] E. Bakolas and P. Tsiotras, "Time-optimal synthesis for the zermelo–markov–dubins problem: the constant wind case," in *American Control Conference (ACC)*, pp. 6163–6168, 2010.

[16] R. L. McNeely, R. V. Iyer, and P. R. Chandler, "Tour planning for an unmanned air vehicle under wind conditions," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1299–1306, 2007.

[17] R. Rysdyk, "Course and heading changes in significant wind," *Journal of guidance, control, and dynamics*, vol. 30, no. 4, pp. 1168–1171, 2007.

[18] L. Techy and C. A. Woolsey, "Minimum-time path planning for unmanned aerial vehicles in steady uniform winds," *Journal of guidance, control, and dynamics*, vol. 32, no. 6, pp. 1736–1746, 2009.

[19] E. Bakolas and P. Tsiotras, "Optimal synthesis of the asymmetric sinistral/dextral markov–dubins problem," *Journal of optimization theory and applications*, vol. 150, no. 2, pp. 233–250, 2011.

[20] H. Choi, *Time-Optimal Paths for a Dubins Car and Dubins Airplane with a Unidirectional Turning Constraint*. PhD thesis, The University of Michigan, 2014.

[21] I. S. Dolinskaya, M. Kotinis, M. Parsons, and R. Smith, "Optimal short-range routing of vessels in a seaway," *Journal of Ship Research*, vol. 53, no. 3, pp. 121–129, 2009.

[22] I. S. Dolinskaya and A. Maggiar, "Time-optimal trajectories with bounded curvature in anisotropic media," *The International Journal of Robotics Research*, vol. 31, no. 14, pp. 1761–1793, 2012.

[23] A. J. Chang, M. Brazil, J. H. Rubinstein, and D. A. Thomas, "Curvature-constrained directional-cost paths in the plane," *Journal of Global Optimization*, vol. 53, no. 4, pp. 663–681, 2012.

[24] G. C. Walsh, R. Montgomery, and S. S. Sastry, "Optimal path planning on matrix lie groups," in *Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on*, vol. 2, pp. 1258–1263, IEEE, 1994.

[25] R. Mukherjee, B. R. Emond, and J. L. Junkins, "Optimal trajectory planning for mobile robots using jacobian elliptic functions," *The International Journal of Robotics Research*, vol. 16, no. 6, pp. 826–839, 1997.

[26] I. Duleba and J. Z. Sasiadek, "Nonholonomic motion planning based on newton algorithm with energy optimization," *IEEE transactions on control systems technology*, vol. 11, no. 3, pp. 355–363, 2003.

[27] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. G. Lee, "Energy-efficient motion planning for mobile robots," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 5, pp. 4344–4349, IEEE, 2004.

[28] M. Moll and L. E. Kavraki, "Path planning for minimal energy curves of constant length," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 3, pp. 2826–2831, IEEE, 2004.

[29] M. Moll and L. E. Kavraki, "Path planning for variable resolution minimal-energy curves of constant length," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2130–2135, IEEE, 2005.

[30] C. H. Kim and B. K. Kim, *Minimum-energy motion planning for differential-driven wheeled mobile robots.* INTECH Open Access Publisher, 2008.

[31] S. C. Spangelo, E. G. Gilbert, A. T. Klesh, P. T. Kabamba, and A. R. Girard, "Periodic energy-optimal path planning for solar-powered aircraft," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2009.

[32] P. Tokekar, N. Karnad, and V. Isler, "Energy-optimal velocity profiles for car-like robots," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 1457–1462, IEEE, 2011.

[33] P. Tokekar, N. Karnad, and V. Isler, "Energy-optimal trajectory planning for car-like robots," *Autonomous Robots*, vol. 37, no. 3, pp. 279–300, 2014.

[34] C. Maclean and J. D. Biggs, "Path planning for simple wheeled robots: sub-riemannian and elastic curves on se (2)," *Robotica*, vol. 31, no. 08, pp. 1285–1297, 2013.

[35] D. Kularatne, S. Bhattacharya, and M. A. Hsieh, “Time and energy optimal path planning in general flows,” in *Proceedings of Robotics: Science and Systems*, (AnnArbor, Michigan), June 2016.

[36] D. T. Greenwood, *Principles of dynamics.* Prentice-Hall Englewood Cliffs, NJ, 1988.

[37] A. Bloch, J. Baillieul, P. Crouch, J. E. Marsden, P. S. Krishnaprasad, R. Murray, and D. Zenkov, *Nonholonomic mechanics and control*, vol. 24. Springer, 2003.

[38] R. F. Gans, *Engineering dynamics.* Springer, 2013.

[39] D. E. Kirk, *Optimal control theory: an introduction.* Courier Corporation, 2012.

[40] L. S. Pontryagin, *Mathematical theory of optimal processes.* CRC Press, 1987.

[41] I. Ross, *A Primer on Pontryagin's Principle in Optimal Control: Second Edition.* Collegiate Publishers, 2015.

[42] F. L. Lewis and V. L. Syrmos, *Optimal control.* John Wiley & Sons, 1995.

[43] D. P. Bertsekas, D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas, *Dynamic programming and optimal control*, vol. 1. Athena Scientific Belmont, MA, 1995.

[44] D. P. Bertsekas, “Dynamic programming and optimal control 3rd edition, volume ii,” *Belmont, MA: Athena Scientific*, 2011.

[45] E. W. Cheney and D. R. Kincaid, *Numerical mathematics and computing.* Nelson Education, 2012.

[46] W. H. Press, *Numerical recipes 3rd edition: The art of scientific computing.* Cambridge university press, 2007.

[47] F. D. Gakhov, *Boundary value problems*, vol. 85. Elsevier, 2014.

[48] U. Ascher and R. D. Russell, “Reformulation of boundary value problems into standard form,” *SIAM review*, vol. 23, no. 2, pp. 238–254, 1981.

[49] R. Bhattacharya, “Optragen: A matlab toolbox for optimal trajectory generation,” in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 6832–6836, IEEE, 2006.

[50] V. Jurdjevic, *Geometric control theory.* Cambridge university press, 1997.